Downloaded from orbit.dtu.dk on: May 05, 2023

DTU Library

Mass Action Stoichiometric. Simulation for Cell Factory Design

Matos, Marta R. A.

Publication date:
2018

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):
Matos, M. R. A. (2018). *Mass Action Stoichiometric. Simulation for Cell Factory Design*. Technical University of Denmark.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Mass Action Stoichiometric Simulation for Cell Factory Design

Marta R. A. Matos

A thesis presented for the degree of
Doctor of Philosophy

The Novo Nordisk Foundation Center for Biosustainability
Technical University of Denmark
2018

Acknowledgements

In no particular order,
Friends and family, without whom nor life nor this PhD would be as enjoyable.
Markus for giving me the opportunity to do this PhD and learning some cool stuff.
Mikael for accepting me as a PhD student, helping with all the paperwork, and valuable feedback.
Niko for all the support and motivation, hard to wish for a better supervisor.
Dan for the great collaboration, for answering all my silly questions, and for bringing me back down to earth when I started questioning everything.
Liz for hosting me at UCSD, all the teachings, and patience.
Novo Nordisk Fonden for PhD funding, as well as Oticon Fond and Otto Mønsted Fond for funding my external stay at UCSD.
Jorge, for helping me out with all the math and statistics, and making me laugh every day.
Alex for always reminding me that models often fail.
Everyone who made my external stay in San Diego a blast, in particular Liz, Dan, Patrick, Kumi, and Cristal.
Ida for not letting me be the last one to finish lunch every single time, and of course, together with Kristian, for the precious help with the abstract's danish version ;)
Moritz, Ravn, Svetlana, for dragging me out of the office on Friday's evenings.
Gheorghe for having my back all the time.
Christian, Kristian, João, Pasquale, Pankaj, Melanie for a fun office (I might miss you, not too sure though...).
Lars for changing my mind about rats.
Julie for the walks.
João for all the help with the not so little things to do before thesis submission.
All the SIMs and everyone else at CfB who made my life easier and happier :)

Last but not least, those two high-school physics teachers, Margarida and Graça, the former for lending me a certain book about Einstein's relativity theory, and the latter for throwing me into research.
It's been a fun ride, thank you all!

Preface

This PhD thesis is written in partial fulfillment of the requirements for the degree of Doctor of Philosophy at the Technical University of Denmark. The work presented here was conducted between the 1st of October of 2014 and the 30th of September of 2017. It was supervised by Professor MSO Mikael Rørdam Andersen, Professor Markus J. Herrgård, and Senior Researcher Nikolaus Sonnenschein. The PhD was funded by the Novo Nordisk Foundation and external funding. The external research stay at the Systems Biology Research Group at the University of San Diego, California was partially funded by the Oticon Fond, the Otto Mønsted Fond, and DTU.

Marta Matos,
30th September 2017
The Novo Nordisk Foundation Center for Biosustainability
Technical University of Denmark

Abstract

For a long time microorganisms have been used to produce beer and bread, and in the last century also molecules such as penicillin and insulin. These same microorganisms can potentially be used to produce a diverse range of other molecules and contribute to a more sustainable future by reducing our dependency on oil. Producing a given molecule at a yield high enough to be commercially viable, however, usually requires cell metabolism to be modified extensively. Traditionally, these modifications have been introduced through random mutagenesis and selection, which has been gradually complemented by more targeted genetic engineering approaches that rely more and more also on computational models of cell metabolism for target selection. Two main types of models can be used here, stoichiometric models or kinetic models. The former are easily built at genome-scale and assume the cell to be in a steady-state, giving information only about the reactions' fluxes, while the latter take into account enzyme dynamics which makes it possible to model substrate-level enzyme regulation and get information about metabolite concentrations and reaction fluxes over time, although at the cost of introducing more parameters. Kinetic models have been plagued by the lack of kinetic data.

The focus of this thesis are kinetic models of cell metabolism. In this work we start by developing a software package to create a model ensemble for individual enzymes in metabolism, where we decompose each reaction into elementary steps, using mass action kinetics to model each step. The resulting rate constants are then fitted to kinetic data (k_{cat}, K_m, K_i, etc.). We then use the package as the basis to build a system-level kinetic model. To do so, we take two different approaches, and in both we drop the assumption that $x_{free} \approx x_{tot}$, i.e. that the total concentration of metabolite in the cell is approximately the same as the free concentration. In both approaches preliminary results show that the fraction of bound metabolite in the cell is not negligible, with some metabolites having an enzyme-bound concentration up to 40%. Next, we address the issue of kinetic data scarcity by using molecular dynamics simulations to estimate the difference in binding energies, $\Delta\Delta G$, between

substrate(s) and a given enzyme and product(s) and the same enzyme for a chosen reaction. Here, we show that these *in silico* determined $\Delta\Delta G$ significantly reduce the amount of rate constants combinations allowed in each model ensemble. Finally, we combine a kinetic model of glycolysis in *Saccharomyces cerevisiae* with time-resolved NMR experiments to study the cellular response to a glucose pulse, and show the model simulations to be in agreement with the experimental results.

Sammenfatning

Mikroorganismer har længe været anvendt til produktion af øl og brød, og gennem det seneste århundrede også til fremstilling af molekyler som penicillin og insulin. Disse mikroorganismer kan potentielt også bruges til at producere en bred vifte af andre molekyler, og derved bidrage til en mere bæredygtig fremtid ved at reducere vores afhængighed af olie. At producere et givent molekyle med et udbytte der er højt nok til at være kommercielt levedygtigt, kræver dog som regel at cellens metabolisme modificeres radikalt. Traditionelt er disse modifikationer blevet udført ved hjælp af tilfældig mutagenese og selektion, hvilket gradvist er blevet komplementeret af mere målrettede genteknologiske metoder, der samtidig afhænger mere og mere af computermodeller af metabolismen til at udvælge targets. Der er to hovedtyper af modeller, der kan anvendes i denne sammenhæng: støkiometriske modeller og kinetiske modeller. Førstnævnte kan let konstrueres på genom-skala og antager at cellen er i steady-state. Dette giver kun information om reaktionernes flux, hvorimod sidstnævnte medregner enzymdynamik, hvilket gør det muligt at modellere enzymregulering på substratniveau, og få information om metabolitkoncentrationer og reaktionsflux over tid. Kinetiske modeller har dog længe været plaget af mangel på kinetiske data.

Fokus for denne afhandling er kinetiske modeller af metabolismen i celler. Vi har udviklet en softwarepakke til at lave et ensemble af modeller for individuelle metaboliske enzymer. Hver reaktion blev opdelt i elementære trin ved hjælp af mass action kinetik til at modellere hvert trin. De resulterende hastighedskonstanter kunne herefter fittes til kinetisk data (k_{cat}, K_m, K_i, etc.). Herefter anvendte vi denne pakke som grundlag for at konstruere en systemisk kinetisk model. Til dette anvendte vi to forskellige fremgangsmåder, hvor vi i begge tilfælde udelod antagelsen at $x_{fri} \approx x_{total}$, altså at den totale koncentration af metabolit i cellen tilnærmelsesvis er lig den frie koncentration. Ved begge fremgangsmåder viste foreløbige resultater at fraktionen af bundet metabolit i cellen ikke er ubetydelig, hvor nogle metabolitter havde en enzymbunden koncentration på op til 40% at den totale koncentration. Herefter

arbejdede vi på at løse problemet med mangel på kinetisk data, ved at anvende molekylære dynamiske simuleringer til at estimere forskellen i bindingsenergi, $\Delta\Delta G$, mellem substrat(er) og et givent enzym samt mellem produkt(er) og det samme enzym for en udvalgt reaktion. Vi viser her, at denne in silico-bestemte $\Delta\Delta G$ signifikant reducerede antallet af kombinationer af hastighedskonstanter der er tilladt i et model-ensemble. Endelig kombinerede vi en kinetisk model for glykolysen i Saccharomyces cerevisiae med time-resolved NMR eksperimenter for at studere den cellulære respons på en glukose-puls, samt for at vise at model simuleringerne var i overensstemmelse med de eksperimentelle resultater.

List of publications

Publications included in this thesis:

- Daniel C. Zielinski*, Marta R.A. Matos*, James E. de Bree, Nikolaus Sonnenschein, Bernhard O. Palsson. *Bottom-up parameterization of enzyme rate constants using corrected enzyme kinetic data* (in preparation).
- Marta R. A. Matos, Daniel C. Zielinski, Bernhard O. Palsson, Nikolaus Sonnenschein, *Elementary Mass Action Stoichiometric Simulation models predict non-negligible fractions of enzyme-bound metabolites* (in preparation).
- Marta R. A. Matos, Nathan Mih, Daniel C. Zielinski, Bernhard O. Palsson, Nikolaus Sonnenschein, Elizabeth Brunk. *Using molecular dynamics simulations to parameterize models of enzyme reaction kinetics* (in preparation).
- Pernille R. Jensen, Marta R. A. Matos, Nikolaus Sonnenschein, and Sebastian Meier. *Reaction control is altered on the seconds timescale in pre-steady state glycolysis: a rapid injection NMR study and its computational simulation* (ready for submission).

Publication not included in this thesis:

- Michele Monti, Marta R. A. Matos, Jeong-Mo Choi, Michael S. Ferry, Bartlomiej Borek. *A modified galactose network model with implications for growth.* arXiv:1601.03235 [q-bio.MN]

Contents

Nomenclature

K_i Inhibiton constant

K_m Half-saturation constant/Michaelis-Menten constant

k_{cat} Turnover constant

13DPG 1,3-biphospho-D-glycerate

23DPG 2,3-diphospho-D-glycerate

2PG 2-phospho-D-glycerate

3PG 3-phospho-D-glycerate

ABC Genetic Regulatory Networks

ADP adenosine-diphosphate

ATP adenosine-triphosphate

DHAP Dihydroxiacetone phosphate

E4P D-erythrose 4-phosphate

EM Ensemble Modeling

eMASS elementary Mass Action Stoichiometric Simulation

ENO Enolase

F6P D-fructose-6-phosphate

FBA Flux Balance Analysis

FBA1 Fructose-biphosphate aldolase, enzyme 1

FBA2 Fructose-biphosphate aldolase, enzyme 2

FDP -D-fructose 1,6-bisphosphate

G3P D-glyceraldehyde 3-phosphate

GAPD Glyceraldehyde-3-phosphate dehydrogenase

GAPDH Glyceraldehyde-3-phosphate dehydrogenase

GRASP General Reaction Assembly and Sampling Platform

LMA Levenberg-Marquardt Algorithm

MASS Mass Action Stoichiometric Simulation

MASSef Mass Action Stoichiometric Simulation enzyme fitting

MWC Monod-Wyman-Changeux

NAD nicotinamide adenine dinucleotide

NADH nicotinamide adenine dinucleotide reduced

NMR Nuclear Magnetic Resonance

ODE Ordinary Differential Equations

ORACLE Optimization and Risk Analysis of Complex Living Entities

PEP phosphoenolpyruvate

PFK1 6-phosphofructokinase I

PGK Phosphoglycerate kinase

PGMd 2,3-bisphosphoglycerate-dependent phosphoglycerate mutase

PGMi 2,3-bisphosphoglycerate-independent phosphoglycerate mutase

PI phosphate

PSO Particle Swarm Optimization

PYK1 Pyruvate kinase I

PYR pyruvate

S7P D-sedoheptulose 7-phosphate

ssld sum of squared log deviations

TALB Transaldolase B

TPI Triose-phosphate isomerase

Chapter 1

Introduction and synopsis

With the widespread use of high throughput technologies that allow for, e.g. the sequencing of a cell's whole genome or the measurement of its metabolome, a more holistic approach is needed to understand the cell's inner workings. Systems biology is such an approach, which aims at studying the whole system rather than studying each component individually. To make sense of the large amount of data produced by omics technologies, computational models are often used. A common approach to study the interactions between different cellular components is to model the system as a network. Three major types of networks can be distinguished: 1) gene regulatory networks, capturing the interactions between transcription factors and genes, 2) signaling networks, modeling signal transduction from cellular receptors to transcription factors, and 3) metabolic networks, encompassing the biochemical reactions that convert various substrates into products. All three networks are encoded in the genome of a cell. Furthermore, all three networks are interconnected, since extracellular signals received by the cell are carried down to transcription factors which affect gene expression, which in turn will affect protein/enzyme expression, while metabolic reaction products also affect gene regulation by modulating the activity of transcription factors. Due to computational limitations, these networks are generally studied separately, although different approaches have been developed to study the interactions between some of these networks [7, 32, 35], and a whole cell model has been recently created for *Mycoplasma genitalium* [25]. In this work we will focus on metabolic networks.

Cell metabolism encompasses the set of biochemical reactions that transform substrates into products. These reactions are catalyzed by enzymes which transform, for instance, sugars into the various building blocks of a cell, e.g. nucleotides, amino acids, and lipids. Understanding cellular metabolism is key to understanding a wide range

of diseases, e.g. the high glycolytic rate exhibited by cancer cells under aerobic conditions commonly known as the Warburg effect [72] or the various monogenic disorders described in Garrod's 'Inborn errors of metabolism [19]. Understanding metabolism, however, is not only important to cure disease but also holds the promise to a more sustainable world, by engineering microorganisms to produce the myriad of chemicals we rely on.

Microorganisms have long been used to produce beer or make bread, and during the past century they have also been adopted to produce pharmaceuticals such as penicillin or insulin. However, to produce non-native molecules or increase yields to a point where they become commercially viable, the cell metabolism needs to be manipulated. While in the past this was done mostly through random mutagenesis, now a more rational strain design approach is followed where it is common to use computational models to guide experiments and develop industrial strains [2, 34].

1.1 Computational models for cell metabolism

Computational models of cell metabolism can be divided in two main types: stoichiometric models and kinetic models. Stoichiometric models consist basically of the set of biochemical reactions encoded by the organism's genome. To build such a model all we need is the organism's genome and respective annotation to know which enzymes can in principle be expressed by the cell. Once the model is built, we can use constraint-based methods to predict phenotypic states using the model. A key assumption used in stoichiometric models is that the system is in a steady-state, i.e. metabolite and enzyme concentrations are constant over time, therefore we can only get information about the metabolic fluxes through the model's reactions. A commonly used constraint-based method is Flux Balance Analysis [14, 39], where an objective function is either maximized or minimized subject to the steady-state condition and, if known, lower and upper bounds on the reactions' fluxes. The objective function typically represents cell growth or ATP production. Other methods, such as MOMA [64] or ROOM [65], can also be used to assess the impact of gene knockouts.

Stoichiometric models can be used to improve cell factory performance [43, 2, 34], as well as to understand disease mechanisms [38, 43, 49] and predict drug targets [29, 36]. However, while these models are relatively easy to build at genome scale, they don't usually account for substrate-level enzyme regulation, which tends to lead to

erroneous predictions, often overproduction of the molecule of interest [27], as regulation imposes a lower upper limit on reactions' fluxes. Approaches to take into account substrate-level enzyme regulation have recently been developed [42, 56], however, a more straightforward way to account for substrate-level enzyme regulation is to use kinetic models.

Kinetic models are built on top of stoichiometric models by taking into account the dynamics of the system, i.e. changes in metabolite concentrations and reactions' fluxes along time, by modeling the kinetics of the catalyzing enzymes. A kinetic model has the general form

$$\frac{d\mathbf{x}}{dt} = \mathbf{S}\,\mathbf{v}(\mathbf{E}, \mathbf{k}, \mathbf{x}, \mathbf{K}_{eq})$$

where $\mathbf{S}$ is the stoichiometric matrix encoding reaction stoichiometry and $\mathbf{v}$ are rate laws that model each reaction's flux. Rate laws $\mathbf{v}$ are usually functions of metabolite and enzyme concentrations ($\mathbf{x}$ and $\mathbf{E}$) as well as enzyme-dependent kinetic parameters $\mathbf{k}$. Sometimes they also include thermodynamic information in the form of equilibrium constants $\mathbf{K}_{eq}$ that are independent of the respective enzymes.

Michaelis-Menten rate laws [44] have been widely used to model metabolism, as these account for enzyme saturation and, for reactions without inhibitors or activators, depend only on either two or three parameters. Either the turnover constant, k_{cat}, the half saturation constant, K_m, and the enzyme concentrations, E_{tot}, are measured explicitly or the saturating flux, $V_{max} = [E_{tot}] \cdot k_{cat}$ and the half saturation constant, K_m, are measured instead. The enzyme concentration is typically a constant in metabolic models. These parameters are fairly easy to characterize *in vitro*. If the enzyme is known to be inhibited or activated by other metabolites, additional inhibition/activation constants and inhibitor/activator metabolite concentrations need to be measured as well. Yet, Michaelis-Menten kinetics do not account for allostery, and the Monod-Wyman-Changeux (MWC) [48] or the Koshland, Nemethy and Filmer (KNF) [31] models need to be used, e.g. by coupling Michaelis-Menten rate law with the MWC model following the generalized MWC model [53]. Recently, a more general type of rate law, convenience kinetics [41], was proposed. These rate laws have similar form to Michaelis-Menten rate laws but model enzyme allostery while requiring a similar amount of parameters and being thermodynamically consistent. Due to kinetic data scarcity, however, rate law parameterization is one of the major bottlenecks in kinetic model construction. Over the years, other simplified rate laws

have been developed that still model the enzyme kinetics while relying on less parameters, examples are log-lin [21], lin-log [71], S-systems [62], or generalized mass action, however, these do not necessarily satisfy thermodynamic constraints. On the other hand, mechanistic approaches have recently been used to build more detailed kinetic models [69, 57, 58, 28] at the cost of introducing more parameters.

1.2 Kinetic modeling approaches

Independently of the chosen rate laws, approaches to build a kinetic model can be roughly divided into top-down or bottom-up approaches. In a bottom-up approach, each enzyme reaction is parameterized individually and all enzymes are then combined to form a systems-level model, while in a top-down approach the whole system is parameterized globally instead of reaction by reaction. Bottom-up approaches typically rely on *in vitro* measured parameters, k_{cat}, K_m, K_i, to parameterize the model's rate laws. This approach has two main drawbacks though, as *in vitro* measurements are often performed under non-physiological conditions thus often not reproducing the *in vivo* behavior, and different parameters are often measured under different conditions, e.g. different pH or temperature, therefore hampering a correct description of the *in vivo* system. Teusink *et al.* [68] built a kinetic model for glycolysis in *Saccharomyces cerevisiae* by gathering kinetic parameters from the literature which were measured *in vitro*, and showed the model was not always able to reproduce the *in vivo* behavior.

To alleviate this issue, kinetic parameters should be measured in conditions that mimic the *in vivo* conditions as closely as possible [70, 13, 66]. Another option is to adjust the kinetic parameters to reproduce the *in vivo* behavior [54], in a way similar to a top-down approach. Top-down approaches avoid incompatibilities between kinetic parameters by fitting the kinetic parameters to omics data, thus in principle, modeling the *in vivo* behavior better. However, these approaches require a large amount of data and are computationally challenging, as they require fitting algorithms capable of fitting dozens of parameters at once. Also, identifiability issues arise [18] when insufficient data is available, which may lead to overfitting. Nevertheless, kinetic models built in a top-down fashion can still produce accurate predictions without precise parameters [20].

In reality though, models are often built by combining bottom-up and top-down approaches. For instance, when using Michaelis-Menten rate laws to build a model it is common to include the kinetic parameters whose values are known and then fit

the remaining parameters simultaneously to omics data, or to simply re-adjust the experimental kinetic parameters to omics data [26, 8, 46].

Several approaches/frameworks have been proposed to build kinetic models over the years, examples are MASS [24], ORACLE [47], Ensemble Modeling (EM) [69], and GRASP-ABC [59, 57].

In the MASS (Mass Action Stoichiometric Simulation) framework, mass action kinetics are used to model each reaction, where the reverse rate constant is substituted by the reaction's equilibrium constant, and fluxomics and metabolomics data are used to calculate the forward rate constant, a pseudo-elementary rate constant (PERC). This approach cannot predict network states too far from the initial state, however, it is scalable to genome scale.

ORACLE, Optimization and Risk Analysis of Complex Living Entities, is another approximate yet scalable approach,. ORACLE relies on stoichiometric and thermodynamics constraints plus metabolomics and fluxomics data, to build a population of models that are thermodynamically and stoichiometrically consistent by sampling metabolite concentrations and enzyme elasticities.

Ensemble modeling (EM), as described in [69], on the other hand, uses elementary reaction mass action kinetics, where each reaction is decomposed in elementary steps according to its mechanism. Here, rate constants are defined in terms of reversibilities and scaled enzyme form concentrations, i.e. the total enzyme concentration measured experimentally is decomposed into free enzyme and enzyme bound to different substrate(s)/product(s). By sampling reaction reversibilities and scaled enzyme form concentrations, different sets of rate constants are obtained that reproduce the original experimental data equally well, resulting in a model ensemble. This model ensemble is then reduced by adding more experimental data. By considering a set of models that reproduce the original data similarly well, the EM formalism avoids overfitting and is able to predict metabolic states far from the initial one, unlike MASS or ORACLE. EM was applied in to build kinetic models for *Escherichia coli*. However, more generally, a model ensemble may refer simply to a set of models with the same structure and different parameter values. For instance, in [37, 26] model ensembles were built by sampling different parameter values for each model to address parameter identifiability issues.

Recently, Saa and Nielsen [59, 57], introduced the GRASP, General Reaction and Assembly Platform, framework combined with Approximate Bayesian computation (ABC). In this approach GRASP is used to generate a set of stoichiometrically and thermodynamically consistent models as priors for the ABC optimization part.

GRASP does so by decomposing each reaction flux as the product of a catalytic and a regulatory function, where the catalytic function is built in a similar way to the EM formalism described above while the regulatory function is based on the generalized MWC model [53]. Once the set of model candidates is produced, the ABC approach is used to produce a set of models that fit the experimental omics data.

Recent approaches have combined kinetic information with stoichiometric models in order to improve predictions. For instance, k-OptForce [11] includes kinetic data for given reactions in the stoichiometric model to improve model predictions, while GECKO [60] integrates k_{cat}s and enzyme concentrations for each reaction in the stoichiometric model by defining an extra constraint: $v \leq k_{cat}\,[E]$. This leads to improved predictions and a decrease in flux variability in more than 60% of the reactions.

1.2.1 Software for kinetic modeling

Different software/web applications are available to build, simulate, and analyze kinetic models. Perhaps the most widely used software application being COPASI [22]. With COPASI we can build, simulate, and analyze kinetic models. When building a model we can use predefined rate laws and estimate parameters using the methods provided, which include genetic algorithms, particle swarm, derivative based methods such as the Levenberg-Marquardt algorithm, among others. The same methods used for parameter estimation, can also be used for general optimization tasks. Methods available for model analysis include Metabolic Control Analysis (MCA), sensitivity analyses of which MCA if part of, time-scale decomposition analysis, calculation of Lyapunov exponents, linear noise approximation to estimate the covariance between particle numbers of different species, etc. Furthermore, COPASI is able to perform both deterministic and stochastic model simulations, as well as a hybrid approach where only some reactions are simulated stochastically. COPASI not only has a friendly graphical interface, but can also be run from the command line and provides bindings for different programming languages, e.g. Python or Java. PySCeS [50] is another software package implemented in Python suited for model simulation, structural and bifurcation analysis, which also supports MCA, and parameter scans. While both COPASI as PySCeS are stand-alone tools that run offline in the user's computer, JWS Online [51] is a user friendly web application that provides most features available in PySCeS but online, without the need of installing a program. Besides, JWS allows one to build their own model and includes a database of curated

models. Yet, note that the model database included in JWS is not as comprehensive as the Biomodels Database . More focused on parameter estimation and identifiability analysis is the AMIGO toolbox [3] which is implemented in Matlab. On the same line, SensA [15] is a web application that allows the user to upload an SBML model and do sensitivity analysis, from local to time-dependent sensitivity analysis. While the above mentioned software solutions focus mostly on model simulation and analysis, CellDesigner [16] focuses on the process of building the model. It allows the user to draw a model on a canvas which is then converted into an SBML model. One can also add rate laws and mathematical equations to these models, which can also be simulated through CellDesigner. Additionally CellDesigner supports parameter search as well. Another application aimed at helping non-expert users create kinetic models is SYCAMORE [73]. It is a web-based application and interfaces with BRENDA [52] and SABIO-RK [74] (both are databases that contain kinetic data) to obtain the required kinetic parameters and equations to build a kinetic model. Moreover, it provides access to parameter estimation based on protein structure [17]. Once the model is built sensitivity analysis can be performed, and the model can be simulated through COPASI on the SYCAMORE server.

All software/web applications mentioned above are compatible with SBML [23], the standard format to store systems biology models. Note that there are many more software solutions for kinetic modeling than the ones mentioned above though [12, 10, 61, 63, 55, 73].

1.3 Thesis motivation and organization

Motivation

Traditional bottom-up kinetic models are usually based on approximate rate laws whose parameters are estimated through *in vitro* experiments, which do not always reproduce the *in vivo* measured data without parameter adjustment [68, 54, 13, 66, 46]. While top-down kinetic models generally suffer from parameter identifiability issues, resulting in "sloppy" parameters [20, 58]. Therefore, in this thesis we propose a new formalism to build kinetic models that bridges both approaches and aims at a more accurate description of the metabolism dynamics.

We first propose a formalism to model each enzymatic reaction individually which avoids assumptions such as the quasi-steady-state or rapid equilibrium assumptions underlying Michaelis-Menten kinetics (see Chapter 2). In this formalism we model

each enzymatic reaction individually by decomposing the reaction into elementary steps according to the enzyme mechanism and generating a mass action rate law for each step. Thus, we avoid implicit assumptions regarding the enzyme mechanism. This results in several elementary rate constants $\boldsymbol{k}$, which are then fitted to kinetic data obtained from literature, e.g. k_{cat}, K_m, K_i, while being constrained by the respective Haldane relation [1] and equilibrium constant. This ensures that the generated rate constants are stoichiometrically and thermodynamically consistent. We term this formalism elementary MASS (eMASS), as it is based on the MASS formalism [24] but decomposes all reactions into elementary steps and not just regulatory reactions. Following this formalism we can also account for kinetic data measured under different conditions, e.g. different cosubstrate concentrations, pH, or temperature. Finally, because the kinetic data used to fit the model can be reproduced equally well by different elementary rate constant $\boldsymbol{k}$ sets, we end up not with a single model but with a model ensemble per enzymatic reaction.

In Chapter 3, we build a prototype model ensemble for eight enzymes in *Escherichia coli*'s glycolysis. We take two different approaches there and compare the results. In a first approach we integrate fluxomics data with each enzyme-level kinetic model generated through eMASS to get fully functional enzyme-level kinetic models. We then define the system-level model to be the combination of the enzyme-level kinetic models. However, for some metabolites, this resulted in a non-negligible amount of metabolite bound to the enzyme, i.e. the assumption $x_{free} \approx x_{tot}$ typically made when building kinetic models, does not necessarily hold. Free metabolite concentrations are represented by x_{free}, while x_{tot} represents the total metabolite concentration in the cell (both free and bound to enzymes). Knowing that mass spectrometry techniques currently used to measure absolute metabolite concentrations cannot distinguish free intracellular metabolites from enzyme-bound metabolites [5, 4, 26, 6], we develop a new approach to integrate the enzyme-level models that distinguishes free metabolite from total metabolite concentrations. Hence, we take the enzyme-level models, define the flux distribution, and sample total enzyme and free metabolite concentrations that reproduce the systems-level fluxomics data. Thus, starting from a base ensemble of models for each enzymatic reaction, this approach is able to fit different phenotypes simply by integrating the respective fluxomics data. Preliminary results show that up to 40% of the total metabolite concentration can be bound to enzyme, a finding that raises concerns about the commonly used assumption that the fraction of bound metabolite is negligible in metabolomics data.

Next we address the issue of kinetic data scarcity in Chapter 4. In particular, we use MM-PBSA [67, 30] to estimate the difference in binding energies, $\Delta\Delta G$, between a given enzymatic reaction's product(s) and substrate(s). We did this for three enzymes in *E. coli*'s central carbon metabolism and integrated the resulting $\Delta\Delta G$ with the respective enzyme-level kinetic models. We show a considerable reduction in the number of combinations of elementary rate constants allowed for each model ensemble. The same approach can be used to integrate the effect of enzyme point mutations in kinetic models [45].

Finally, in Chapter 5, we collaborated with experimental scientists to study the cellular response of *Saccharomyces cerevisiae* to different glucose pulses. Metabolite time-courses were measured by hyperpolarized NMR spectroscopy in three different experiments: a) a glucose pulse, b) a glucose + acetaldehyde pulse, and c) a glucose + ethanol pulse experiment. We used a published model for *S. cerevisiae*'s glycolysis [66] to simulate the pulse experiments and compare the accumulation in intermediate glycolytic metabolites to the change in the NAD^+/NADH ratio. The simulation results, consistent with the experiments and the literature, corroborate the hypothesis that accumulation of different metabolites may be driven by changes in the NAD^+/NADH ratio.

Thesis outline

This thesis is divided into the following chapters:

Chapter 2 Describes eMASS and respective implementation as a software package, MASSef. It is based on the article: Daniel C. Zielinski*, Marta R.A. Matos*, James E. de Bree, Nikolaus Sonnenschein, Bernhard O. Palsson. *Bottom-up parameterization of enzyme rate constants using corrected enzyme kinetic data* (in preparation).

Chapter 3 Construction of a prototype model ensemble using eMASS and eMASS2 which address free vs. total metabolite concentrations. It is based on the article: Marta R. A. Matos, Daniel C. Zielinski, Bernhard O. Palsson, Nikolaus Sonnenschein, *Elementary Mass Action Stoichiometric Simulation models predict non-negligible fractions of enzyme-bound metabolites* (in preparation).

Chapter 4 Integration of relative binding energies estimated through molecular dynamics with enzyme-level kinetic models. It is based on the article: Marta R. A. Matos, Nathan Mih, Daniel C. Zielinski, Bernhard O. Palsson, Nikolaus Sonnenschein,

Elizabeth Brunk. *Using molecular dynamics simulations to parameterize models of enzyme reaction kinetics* (in preparation).

Chapter 5 Integration of NMR data with a kinetic model of *S. cerevisiae* glycolysis to study the cellular response to different glucose pulses. It is based on the article: Pernille R. Jensen, Marta R. A. Matos, Nikolaus Sonnenschein, and Sebastian Meier. *Reaction control is altered on the seconds timescale in pre-steady state glycolysis: a rapid injection NMR study and its computational simulation* (ready for submission).

Chapter 6 Summary of the work done in this thesis, discussion of the difficulties faced in each project and presentation of possible solutions and future work.

References

[1] R. A. Alberty. The relationship between michaelis constants, maximum velocities and the equilibrium constant for an enzyme-catalyzed reaction. eng. *Journal of the american chemical society*, 75(8):1928–1932, 1953.

[2] J. Almquist, M. Cvijovic, V. Hatzimanikatis, J. Nielsen, and M. Jirstrand. Kinetic models in industrial biotechnology - Improving cell factory performance. *Metabolic engineering*, 24C:38–60, 2014.

[3] E. Balsa-Canto, D. Henriques, A. Gábor, and J. R. Banga. Amigo2, a toolbox for dynamic modeling, optimization and control in systems biology. *Bioinformatics*, 32(21):3357–3359, 2016. ISSN: 14602059. DOI: 10.1093/bioinformatics/btw411.

[4] B. D. Bennett, E. H. Kimball, M. Gao, R. Osterhout, S. J. Van Dien, and J. D. Rabinowitz. Absolute metabolite concentrations and implied enzyme active site occupancy in *Escherichia coli*. *Nature chemical biology*, 5(8):593–9, Aug. 2009.

[5] B. D. Bennett, J. Yuan, E. H. Kimball, and J. D. Rabinowitz. Absolute quantitation of intracellular metabolite concentrations by an isotope ratio-based approach. *Nature protocols*, 3(8):1299–1311, 2008.

[6] A. B. Canelas, W. M. Van Gulik, and J. J. Heijnen. Determination of the cytosolic free NAD/NADH ratio in *Saccharomyces cerevisiae* under steady-state and highly dynamic conditions. *Biotechnology and bioengineering*, 100(4):734–743, 2008.

[7] S. Chandrasekaran and N. D. Price. Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in *Escherichia coli* and *Mycobacterium tuberculosis*. *Proceedings of the national academy of sciences*, 107(41):17845–17850, 2010.

[8] C. Chassagnole, N. Noisommit-Rizzi, J. W. Schmid, K. Mauch, and M. Reuss. Dynamic modeling of the central carbon metabolism of *Escherichia coli*. *Biotechnology and bioengineering*, 79(1):53–73, 2002.

[9] V. Chelliah, N. Juty, I. Ajmera, R. Ali, M. Dumousseau, M. Glont, M. Hucka, G. Jalowicki, S. Keating, V. Knight-Schrijver, A. Lloret-Villas, K. Nath Natarajan, J.-B. Pettit, N. Rodriguez, M. Schubert, S. M. Wimalaratne, Y. Zhao, H. Hermjakob, N. Le Novère, and C. Laibe. BioModels: ten-year anniversary. *Nucl. acids res.*, 43:D542–D548, D1, 2015. DOI: 10.1093/nar/gku1181.

[10] K. Choi, J. K. Medley, C. Cannistra, and K. Matthias. Tellurium : a python based modeling and reproducibility platform for systems biology. *Biorxiv*:1–27, 2016. DOI: 10.1101/054601.

[11] A. Chowdhury, A. R. Zomorrodi, and C. D. Maranas. k-OptForce: integrating kinetics with flux balance analysis for strain design. *Plos computational biology*, 10(2):e1003487, Feb. 2014.

[12] R. Clewley. Hybrid models and biological model reduction with pydstool. *Plos computational biology*, 8(8), 2012. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1002628.

[13] K. v. Eunen, J. A. L. Kiewiet, H. V. Westerhoff, and B. M. Bakker. Testing Biochemistry Revisited: How *In Vivo* Metabolism Can Be Understood from *In Vitro* Enzyme Kinetics. *Plos computational biology*, Jan. 2012.

[14] D. A. Fell and J. R. Small. Fat synthesis in adipose tissue. An examination of stoichiometric constraints. *Biochemical journal*, 238(3):781–786, 1986.

[15] M. Floettmann, J. Uhlendorf, T. Scharp, E. Klipp, and T. W. Spiesser. Sensa: web-based sensitivity analysis of sbml models. *Bioinformatics*, 30(19):2830–2831, 2014. ISSN: 14602059. DOI: 10.1093/bioinformatics/btu378.

[16] B. A. Funahashi, Y. Matsuoka, A. Jouraku, M. Morohashi, N. Kikuchi, and H. Kitano. A versatile modeling tool for biochemical networks. *Proceedings of the ieee*, 96(8):1254–1265, 2008. ISSN: 0018-9219. DOI: 10.1109/JPROC.2008.925458.

[17] R. R. Gabdoulline, M. Stein, and R. C. Wade. Qpipsa: relating enzymatic kinetic parameters and interaction fields. eng. *Bmc bioinformatics*, 8(1):373, 2007. ISSN: 14712105. DOI: 10.1186/1471-2105-8-373.

[18] A. Gábor, A. F. Villaverde, and J. R. Banga. Parameter identifiability analysis and visualization in large-scale kinetic models of biosystems. *Bmc systems biology*, 1154(11):1–16, 2017.

[19] A. Garrod. *Inborn errors of metabolism*. Of *Oxford medical publications*. H. Frowde and Hodder & Stoughton, 1909.

[20] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, and J. P. Sethna. Universally sloppy parameter sensitivities in systems biology models. *Plos computational biology*, 3(10):1871–1878, 2007.

[21] V. Hatzimanikatis and J. E. Bailey. Effect of spatiotemporal variations on metabolic control: Approximate analysis using (log) linear kinetic models. *Biotechnol bioeng*, 54:91–104, 1996.

[22] S. Hoops, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI - A COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, 2006.

[23] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A. A. Cuellar, S. Dronov, E. D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J. H. Hofmeyr, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Nov??re, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. F. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang. The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[24] N. Jamshidi and B. Ø. Palsson. Mass action stoichiometric simulation models: incorporating kinetics and regulation into stoichiometric models. *Biophysical journal*, 98(2):175–85, Jan. 2010.

[25] J. R. Karr, J. C. Sanghvi, D. N. Macklin, M. V. Gutschow, J. M. Jacobs, B. Bolival, N. Assad-Garcia, J. I. Glass, and M. W. Covert. A whole-cell computational model predicts phenotype from genotype. eng. *Cell*, 150(2):389–401, 2012.

[26] D. Kesten, U. Kummer, S. Sahle, and K. Hübner. A new model for the aerobic metabolism of yeast allows the detailed analysis of the metabolic regulation during glucose pulse. *Biophysical chemistry*, 206:40–57, 2015.

[27] A. Khodayari and C. D. Maranas. A genome-scale escherichia coli kinetic metabolic model satisfying flux data for multiple mutant strains. *Nature communications*, 7:1–12, 2016. ISSN: 2041-1723. DOI: 10.1038/ncomms13806. URL: http://dx.doi.org/10.1038/ncomms13806.

[28] A. Khodayari, A. R. Zomorrodi, J. C. Liao, and C. D. Maranas. A kinetic model of Escherichia coli core metabolism satisfying multiple sets of mutant flux data. *Metabolic engineering*, 25, 2014. ISSN: 10967184. DOI: 10.1016/j.ymben.2014.05.014.

[29] E. Klipp, R. C. Wade, and U. Kummer. Biochemical network-based drug-target prediction. *Current opinion in biotechnology*, 21(4):511–516, 2010.

[30] P. A. Kollman, I. Massova, C. Reyes, B. Kuhn, S. Huo, L. Chong, M. Lee, T. Lee, Y. Duan, W. Wang, O. Donini, P. Cieplak, J. Srinivasan, D. A. Case, and T. E. Cheatham. Calculating Structures and Free Energies of Complex Molecules: Combining Molecular Mechanics and Continuum Models. *Accounts of chemical research*, 33(12):889–897, 2000.

[31] D. E. Koshland, G. Némethy, and D. Filmer. Comparison of experimental binding data and theoretical models in proteins containing subunits*. *Biochemistry*, 5(1):365–385, 1966. DOI: 10.1021/bi00865a047.

[32] O. Kotte, J. B. Zaugg, and M. Heinemann. Bacterial adaptation through distributed sensing of metabolic fluxes. *Molecular systems biology*, 6(355):355, Jan. 2010.

[33] N. Le Novère, B. Bornstein, A. Broicher, M. Courtot, M. Donizelli, H. Dharuri, L. Li, H. Sauro, M. Schilstra, B. Shapiro, J. L. Snoep, and M. Hucka. BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucleic acids research*, 34(Database issue):D689–D691, Jan. 2006.

[34] S. Y. Lee and H. U. Kim. Systems strategies for developing industrial microbial strains. *Nature biotechnology*, 33(10):1061–1072, 2015.

[35] J. A. Lerman, D. R. Hyduke, H. Latif, V. A. Portnoy, N. E. Lewis, J. D. Orth, A. C. Schrimpe-Rutledge, R. D. Smith, J. N. Adkins, K. Zengler, and B. O. Palsson. In silico method for modelling metabolism and gene product expression at genome scale. *Nature communications*, 3:929, 2012.

[36] J. Levering, T. Fiedler, A. Sieg, K. W. v. Grinsven, S. Hering, N. Veith, B. G. Olivier, L. Klett, J. Hugenholtz, B. Teusink, B. Kreikemeyer, and U. Kummer. Genome-scale reconstruction of the *Streptococcus pyogenes* M49 metabolic network reveals growth requirements and indicates potential drug targets. *Journal of biotechnology*, 232:25–37, 2016.

[37] J. Levering, M. W. J. M. Musters, M. Bekker, D. Bellomo, T. Fiedler, W. M. De Vos, J. Hugenholtz, B. Kreikemeyer, U. Kummer, and B. Teusink. Role of phosphate in the central metabolism of two lactic acid bacteria - a comparative systems biology approach. *Febs journal*, 279(7):1274–1290, 2012. ISSN: 1742464X. DOI: 10.1111/j.1742-4658.2012.08523.x.

[38] N. E. Lewis and A. M. Abdel-Haleem. The evolution of genome-scale models of cancer metabolism. *Frontiers in physiology*, 4:237, 2013.

[39] N. E. Lewis, H. Nagarajan, and B. O. Palsson. Constraining the metabolic genotype–phenotype relationship using a phylogeny of *in silico* methods. *Nature reviews microbiology*, 10(4):291–305, 2012.

[40] C. Li, M. Donizelli, N. Rodriguez, H. Dharuri, L. Endler, V. Chelliah, L. Li, E. He, A. Henry, M. I. Stefan, J. L. Snoep, M. Hucka, N. Le Novère, and C. Laibe. BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. *Bmc systems biology*, 4:92, June 2010.

[41] W. Liebermeister and E. Klipp. Bringing metabolic networks to life: convenience rate law and thermodynamic constraints. *Theoretical biology & medical modelling*, 3:41, Jan. 2006.

[42] D. Machado, M. J. Herrgård, and I. Rocha. Modeling the Contribution of Allosteric Regulation for Flux Control in the Central Carbon Metabolism of *E. coli*. *Frontiers in bioengineering and biotechnology*, 3(October):1–11, 2015.

[43] D. McCloskey, B. Ø. Palsson, and A. M. Feist. Basic and applied uses of genome-scale metabolic network reconstructions of *Escherichia coli*. *Molecular systems biology*, 9(1):661, 2013.

[44] L. Michaelis and M. Menten. Die kinetik der invertinwirkung. *Biochem z.*, (49):333–369, 1913.

[45] N. Mih, E. Brunk, A. Bordbar, and B. O. Palsson. A multi-scale computational platform to mechanistically assess the effect of genetic variation on drug responses in human erythrocyte metabolism. eng. *Plos computational biology*, 12(7):e1005039, e1005039, 2016.

[46] P. Millard, K. Smallbone, and P. Mendes. Metabolic regulation is sufficient for global and robust coordination of glucose uptake, catabolism, energy production and growth in *Escherichia coli*. *Plos computational biology*, 13(2):e1005396, 2017.

[47] L. Miskovic and V. Hatzimanikatis. Production of biofuels and biochemicals: In need of an ORACLE. *Trends in biotechnology*, 28(8):391–397, 2010.

[48] J. Monod, J. Wyman, and J. P. Changeaux. On the nature of allosteric transitions: a plausible model. eng. *Journal of molecular biology*, 12(1):88–118, 88–118, 1965. ISSN: 10898638, 00222836. DOI: 10.1016/S0022-2836(65)80285-6.

[49] J. Nielsen. Systems Biology of Metabolism : A Driver for Developing Personalized and Precision Medicine. *Cell metabolism*, 25(3):572–579, 2017.

[50] B. G. Olivier, J. M. Rohwer, and J. H. S. Hofmeyr. Modelling cellular systems with pysces. *Bioinformatics*, 21(4):560–561, 2005. ISSN: 13674803. DOI: 10.1093/bioinformatics/bti046.

[51] B. G. Olivier and J. L. Snoep. Web-based kinetic modelling using jws online. *Bioinformatics*, 20(13):2143–2144, 2004. DOI: 10.1093/bioinformatics/bth200.

[52] S. Placzek, I. Schomburg, A. Chang, L. Jeske, M. Ulbrich, J. Tillack, and D. Schomburg. BRENDA in 2017: New perspectives and new tools in BRENDA. *Nucleic acids research*, 45(D1):D380–D388, 2017.

[53] S. V. Popova and E. E. Sel'kov. Generalization of the model by monod, wyman and changeux for the case of a reversible monosubstrate reaction. *Febs letters*, 53(3):269–273, 1975.

[54] L. Pritchard and D. B. Kell. Schemes of flux control in a model of *Saccharomyces cerevisiae* glycolysis. *European journal of biochemistry*, 269(16):3894–3904, Aug. 2002.

[55] A. Raue, B. Steiert, M. Schelker, C. Kreutz, T. Maiwald, H. Hass, J. Vanlier, C. Tönsing, L. Adlung, R. Engesser, W. Mader, T. Heinemann, J. Hasenauer, M. Schilling, T. Höfer, E. Klipp, F. Theis, U. Klingmüller, B. Schöberl, and J. Timmer. Data2dynamics: a modeling environment tailored to parameter estimation in dynamical systems. *Bioinformatics*, 31(21):3558–3560, 2015. ISSN: 14602059. DOI: 10.1093/bioinformatics/btv405.

[56] E. Reznik, D. Christodoulou, J. E. Goldford, E. Briars, U. Sauer, D. Segrè, and E. Noor. Genome-Scale Architecture of Small Molecule Regulatory Networks and the Fundamental Trade-Off between Regulation and Enzymatic Activity. *Cell reports*, 20(11):2666–2677, 2017.

[57] P. A. Saa and L. K. Nielsen. Construction of feasible and accurate kinetic models of metabolism: A Bayesian approach. *Scientific reports*, 6(July):29635, 2016.

[58] P. A. Saa and L. K. Nielsen. Formulation, construction and analysis of kinetic models of metabolism: A review of modelling frameworks. *Biotechnology advances*, 2017.

[59] P. Saa and L. K. Nielsen. A general framework for thermodynamically consistent parameterization and efficient sampling of enzymatic reactions. *Plos computational biology*, 11(4):e1004195, 2015.

[60] B. J. Sánchez, C. Zhang, A. Nilsson, P.-J. Lahtvee, E. J. Kerkhoven, and J. Nielsen. Improving the phenotype predictions of a yeast genome-scale metabolic model by incorporating enzymatic constraints. *Molecular systems biology*, 13(8):935, 2017.

[61] A. Sastry and N. Sonnenschein. MASS-Toolbox: Introducing Annotations and Pathway Attributes. Jan. 2016. URL: `https://doi.org/10.5281/zenodo.44339`.

[62] M. A. Savageau. Biochemical systems analysis. I. Some mathematical properties of the rate law for the component enzymatic reactions. *Journal of theoretical biology*, 25(3):365–369, 1969.

[63] H. Schmidt and M. Jirstrand. Systems Biology Toolbox for MATLAB: A computational platform for research in systems biology. *Bioinformatics*, 22(4):514–515, 2006. ISSN: 13674803. DOI: `10.1093/bioinformatics/bti799`.

[64] D. Segre, D. Vitkup, and G. M. Church. Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the national academy of sciences*, 99(23):15112–15117, 2002.

[65] T. Shlomi, O. Berkman, and E. Ruppin. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. eng. *Proceedings of the national academy of sciences of the united states of america*, 102(21):7695–7700, 2005.

[66] K. Smallbone, H. L. Messiha, K. M. Carroll, C. L. Winder, N. Malys, W. B. Dunn, E. Murabito, N. Swainston, J. O. Dada, F. Khan, P. Pir, E. Simeonidis, I. Spasić, J. Wishart, D. Weichart, N. W. Hayes, D. Jameson, D. S. Broomhead, S. G. Oliver, S. J. Gaskell, J. E. G. McCarthy, N. W. Paton, H. V. Westerhoff, D. B. Kell, and P. Mendes. A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *Febs letters*, 587(17):2832–41, Sept. 2013.

[67] J. Srinivasan, T. E. Cheatham, P. Cieplak, P. A. Kollman, and D. A. Case. Continuum Solvent Studies of the Stability of DNA, RNA, and PhosphoramidateDNA Helices. *Journal of the american chemical society*, 120(37):9401–9409, 1998.

[68] B. Teusink, J. Passarge, C. Reijenga, E. Esgalhado, C. v. d. Weijden, M. Schepper, M. Walsh, B. Bakker, K. v. Dam, H. Westerhoff, and J. Snoep. Can yeast glycolysis be understood in terms of *in vitro* kinetics of the constituent enzymes? testing biochemistry. eng. *European journal of biochemistry*, 267(17):5313–5329, 2000.

[69] L. M. Tran, M. L. Rizk, and J. C. Liao. Ensemble modeling of metabolic networks. *Biophysical journal*, 95(12):5606–17, Dec. 2008.

[70] K. Van Eunen, J. Bouwman, P. Daran-Lapujade, J. Postmus, A. B. Canelas, F. I. C. Mensonides, R. Orij, I. Tuzun, J. Van Den Brink, G. J. Smits, W. M. Van Gulik, S. Brul, J. J. Heijnen, J. H. De Winde, M. J. Teixeira De Mattos, C. Kettner, J. Nielsen, H. V. Westerhoff, and B. M. Bakker. Measuring enzyme activities under standardized in *vivo*-like conditions for systems biology. *Febs journal*, 277(3):749–760, 2010.

[71] D. Visser and J. J. Heijnen. Dynamic simulation and metabolic re-design of a branched pathway using linlog kinetics. *Metabolic engineering*, 5(3):164–176, 2003.

[72] O. Warburg. On the origin of cancer cells. eng. *Science*, 123(3191):309–314, 1956.

[73] A. Weidemann, S. Richter, M. Stein, S. Sahle, R. Gauges, R. Gabdoulline, I. Surovtsova, N. Semmelrock, B. Besson, I. Rojas, R. Wade, and U. Kummer. SYCAMORE - A systems biology computational analysis and modeling research environment. *Bioinformatics*, 24(12):1463–1464, 2008. ISSN: 13674803. DOI: 10.1093/bioinformatics/btn207.

[74] U. Wittig, R. Kania, M. Golebiewski, M. Rey, L. Shi, L. Jong, E. Algaa, A. Weidemann, H. Sauer-Danzwith, S. Mir, O. Krebs, M. Bittkowski, E. Wetsch, I. Rojas, and W. Müller. SABIO-RK - Database for biochemical reaction kinetics. *Nucleic acids research*, 40(D1):790–796, 2012.

Chapter 2

Bottom-up parameterization of enzyme rate constants using corrected enzyme kinetic data

Abstract

Biochemical kinetic models are becoming increasingly useful in solidifying theoretical understanding of biological processes. Due to a number of issues, including nonlinearity and poor scaling, parameterizing these models has generally been difficult. Top-down methods such as parameter sampling and system-level fitting have largely been adopted to deal with data gaps. However, bottom-up methods have the advantage of utilizing the vast amount of measured kinetic data, if challenges such as gaps in data and *in vitro* to *in vivo* differences can be overcome. Here, we introduce eMASS (elementary Mass Action Stoichiometry Simulation), a computational workflow for the robust calculation of rate constants of enzyme-level kinetic models that fit kinetic data while taking into account uncertainty. The eMASS framework is implemented as a software package, termed MASSef (Mass Action Stoichiometry Simulation enzyme fitting), and can handle the majority of standard enzyme parameters, including, K_m, k_{cat}, K_i, K_{eq}, and n_H. Furthermore, the approach can use the media conditions under which the data was measured to handle different co-substrate concentrations and perform certain corrections for ion binding and assay temperature to *in vivo*-like conditions when possible. We provide four case studies demonstrating the approach for various inhibition schemes, enzyme activation, different kinetic behavior, and temperature correction. The code is provided in the MASSef package (https://github.com/martamatos/MASSef) built on top of the MASS Toolbox (http://opencobra.github.io/MASS-Toolbox/) in Mathematica®. The determination of biochemical elementary reaction parameters is a fundamental pursuit in understanding the genotype-phenotype relationship mechanistically and should support further efforts towards prospective modeling of emergent network behavior.

2.1 Introduction

There has been a resurgence of interest in the construction of large-scale kinetic models of metabolism for model organisms in recent years [38, 27, 23]. These models hold promise in a number of applications that other models, such as stoichiometric models, have difficulty addressing, for instance the role of metabolite concentrations and allosteric regulation in the control of metabolic flux [26, 4, 35, 7]. However, the primary issue impeding the development of practical large-scale kinetic models of metabolism is the need for a large number of kinetic parameters, the vast majority of which have not been experimentally measured [18]. These additional parameters arise because kinetic models require specification of reaction rate laws, which generally contain initial metabolite concentrations, enzyme concentrations, reaction equilibrium constants, and various kinetic constants. However, the majority of models are condition specific and include inherent assumptions, making their generalizability and scalability uncertain. Biochemical reactions fundamentally occur via elementary reaction steps with rate laws that are often of a known form but have unknown parameters.

A number of approaches for the parameterization of kinetic models of metabolism have been developed. These approaches can be loosely classified into sampling, top-down, and bottom-up methods, based on the type of data used to parameterize the model. Sampling methods randomly sample parameters in particular expected ranges, sometimes with certain constraints such as thermodynamic consistency enforced in the sampling procedure [33, 28, 41]. Top-down parameterization methods use data on the kinetic behavior of the entire system and parameterize the entire model simultaneously to match this data [6, 41, 23]. Both sampling and top-down methods can efficiently define the large number of parameters required by a kinetic model. Further, a number of software tools are available for parameterization using these methods [1, 15, 19, 14]. However, these methods can suffer from the potential issues related to limited predictability due to either attempting to sample too wide a parameter space in the case of sampling methods, or having too many parameters for the limited amount of data, resulting in overfitting in the case of system-level parameterization. Both of these issues raise problems in the ability of these models to accurately predict system behavior in new conditions. Bottom-up methods on the other hand use data related to the individual components of the network to construct a model piece by piece. Bottom-up methods have the advantage of utilizing the vast amount of historical enzyme data [30, 9]. However, there are a number of difficulties with bottom-up construction of kinetic models that have impeded their

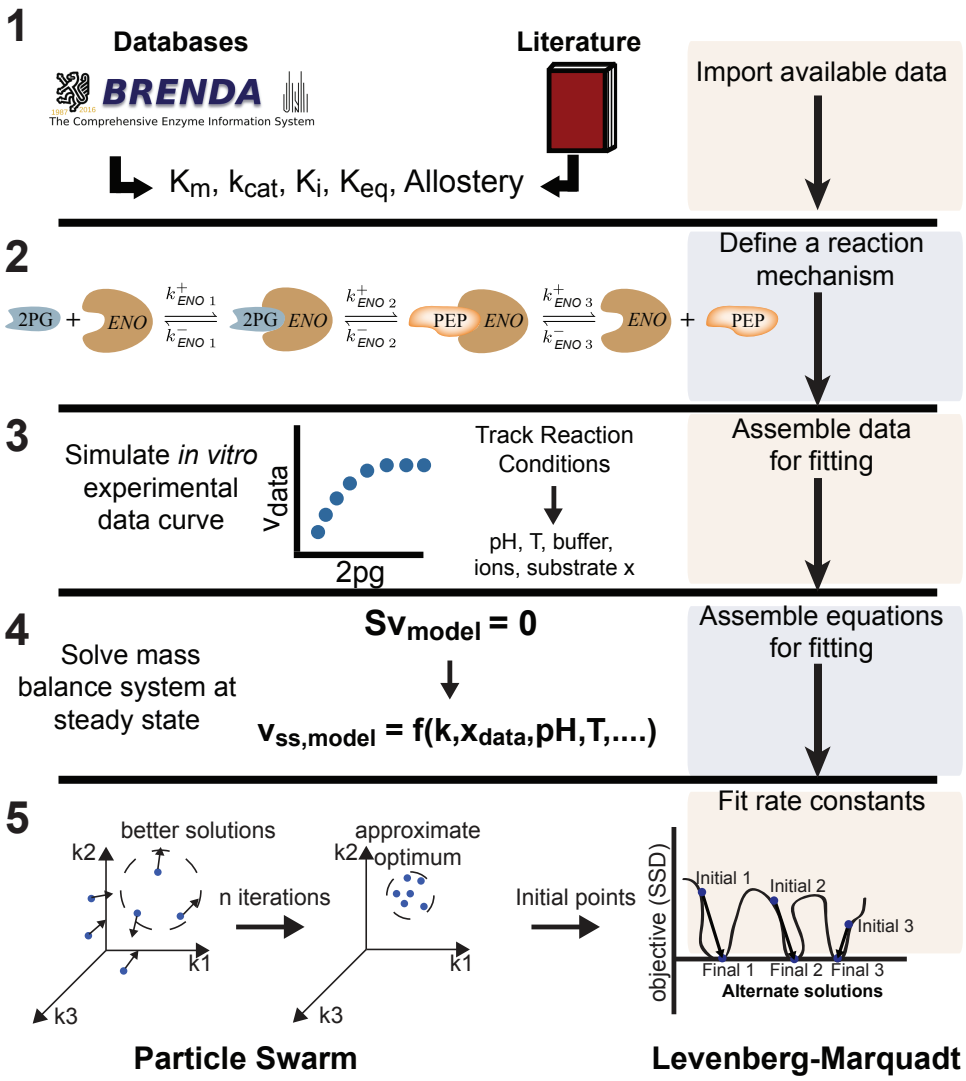
development. First, the majority of enzymes do not have detailed kinetic assays performed to measure the required kinetic parameters to construct such models. As a result, kinetic models are often highly underdetermined and methods to estimate missing parameters must be developed [24]. Also, enzyme kinetics are often measured under non-physiological conditions, raising questions about their relevance to modeling *in vivo* systems [39, 12, 42, 16]. However, recent high-throughput studies have shown substantial correlations between *in vivo* and *in vitro* kinetic parameters, supporting the use of available enzyme kinetic data [10, 5]. Furthermore, modeling studies attempting to build kinetic models from single enzyme information have been fairly successful [20, 42, 37]. Thus, it appears that bottom-up methods may be promising complementary alternatives for the construction of large-scale kinetic models of metabolism if current challenges can be overcome.

Here, we present a computational workflow that attempts to address a number of open issues with bottom-up construction of kinetic models of metabolism. This workflow is based on the Mass Action Stoichiometric Simulation (MASS) framework [21]. However, the focus here is not on modeling each reaction with generalized mass action rate laws and use omics data together with equilibrium constants to parameterize the model. Instead, we parameterize enzyme-level kinetic models where each reaction is decomposed into elementary reactions according to the enzyme mechanism, which are then modeled by a mass action rate law. Therefore we call this workflow elementary MASS (eMASS). By using eMASS we parameterize rate constants in a manner that satisfies a variety of measured enzyme data types, accounts for parameter uncertainty, and allows the correction of data to *in vivo*-like conditions where data is available. This workflow is implemented as a software package in Mathematica® termed MASSef (Mass Action Stoichiometry Simulation enzyme fitting). There are three core features of this software package: 1) a symbolic algebra system that generates equations for comparison despite the complexity of the enzyme mechanism, 2) a data processing procedure that utilizes reported experimental conditions to handle different co-substrate concentrations and allow biophysical corrections to *in vivo*-like conditions where possible, and 3) robust nonlinear optimization to fit the model to data while accounting for uncertainty. We first present an overview of the computational workflow for parameter fitting before discussing the details of individual components, and then present four case studies demonstrating the workflow for enzymes with different available kinetic data and regulation.

2.2 Results

Overview of computational parameter fitting pipeline

eMASS is divided into five key steps, which are shown in Figure 2.1. 1) Gather enzyme data to be used in rate constant fitting and annotate experimental conditions. Data point weights are defined by the user, which determine the weighting of each data point in the fitting procedure. 2) The enzymatic reaction is decomposed into elementary steps by the user according to the respective enzyme mechanism, and a mass action rate law is generated for each elementary reaction. 3) Using the generated rate laws, define equations corresponding to the data points that are to be fitted in step 5. 4) Process the data to be fitted with experimental conditions in a way that allows direct comparison to the equations defined in step 3. 5) Fit the equations defined in step 3 to the processed data in step 4 using a multi-step nonlinear least squares optimization. We repeat this optimization N times and cluster the resulting rate constant sets to try and identify a reduced number of characteristic rate constant sets for the enzyme. Optionally, to build a fully functional enzyme-level ODE model, a final step can be to integrate metabolomics data and either fluxomics data or proteomics data. We will use enolase (ENO) to exemplify some of the steps.



Minimize non-linear least square objective:

$$\min_k \sum(\log(\text{model}) - \log(\text{data}))^2$$

Figure 2.1: Workflow for parameterization of rate constants using enzyme kinetic data. This workflow consists of five steps: 1) Gathering and curating kinetic data, 2) defining an enzyme mechanism, 3) processing kinetic data and correcting it to in vivo-like conditions, 4) defining equations that relate model behavior to each data type, 5) a nonlinear least-squares optimization to identify rate constant sets that fit available kinetic data.

Preparation of enzyme kinetic data

First, kinetic data on enzymes is gathered, curated, and placed into a table in a

Parameter	Value	Reference (PMID)
EC number	4.2.1.11	-
Organism	*E. coli*	-
Reaction	$2PG \rightleftharpoons PEP + H_2O$	-
Mechanism	Null	-
# Subunits	Null	-
#Active sites	2	11676541
#Allosteric sites	2	11676541

(a)

Parameter	Weight	Metabolite	Co-substrate	Value	Uncertainty	Units	T (C)	pH	Buffer	Salts	Reference (PMID)
K_{eq}	1	2PG	-	5.19	[4.9305, 5.4495]	Null	25	7.5	-	-	eQuilibrator
k_{cat}	1	2PG	-	330	[313.5, 346.5]	s^{-1}	30	8.1	trishcl, 0.05	Cl, 0.1; K, 0.1	estimated
K_m	1	2PG	Mg^{2+}, 0.001; SO_4, 0.001	0.0001	[0.000095, 0.000105]	M	30		trishcl, 0.05	Cl, 0.1; K, 0.1	4942326

(b)

Table 2.1: (a) Data relative to the reaction overall and the enzyme structure. (b) Available kinetic data for ENO and conditions in which it was measured.

standard format. Data types currently handled include enzyme structure, reaction stoichiometry, reaction mechanism, reaction equilibrium constant, K_{eq}, dissociation constants, K_d, and standard initial rate kinetic assay constants such as K_m, k_{cat}, and K_i. For reaction equilibrium constants and initial rate constants, experimental data such as media conditions, pH, and temperature are also extracted for possible use in data adjustments to *in vivo*-like conditions, as discussed later. Data point weights are set in a continuous range as determined by the user. This allows the user additional control over how heavily to consider data that may be conflicting with other data measurements or have other reliability issues. Additionally, data point weights can be used to distribute the error between the measured data points and the values predicted by the model more evenly. Instead of, e.g. fitting data points with high absolute values better than data points with low absolute values. This will be illustrated later in the case studies.

For ENO the available data is shown in Table 2.1.

Specify the enzyme mechanism

A mechanism for the enzymatic reaction is to be specified by the user, including

binding of substrates to the enzyme, conversion, and product release. As these reaction steps do not proceed through a single transition state, we do not consider them to be true elementary reactions, but instead, and according to Cleland's nomenclature [8], microscopic reaction steps with microscopic kinetic rate constants. However, to be consistent with the literature in the field, we will still refer to them as elementary reactions [32].

The method is flexible to various reaction schemes, such as binding order, bi-bi mechanisms, ping-pong mechanisms, and slow enzyme transitions. Both reversible and irreversible reactions can be specified, but fully reversible mechanisms are recommended for later use of thermodynamic Haldane relationships [3]. Conversion steps are typically represented as a single transition between reactants and products; however, in principle more detailed mechanisms involving individual reaction steps of catalytic residues could also be modeled. Generally, protons and water are assumed constants and are excluded from reaction mechanisms. At this stage, individual catalytic pathways and thermodynamic cycles can be defined by the user to serve as Haldane thermodynamic constraints, arising from the First Law of Thermodynamics. A catalytic pathway consists of a particular set of catalytic reactions that convert substrates into products. Haldane constraints, which correspond to pathways from the substrate to the product, are later fitted based on the reaction K_{eq}. Thermodynamic cycles with the enzyme are equivalent catalytic sub-pathways within the enzyme mechanism that must be balanced and thus reduce the number of degrees of freedom of the rate constants by one.

In addition, substrate-level enzyme inhibition and activation can be modeled by adding the elementary reactions corresponding to the respective mechanism. For enzyme activation, competitive, uncompetitive, and mixed inhibition mechanisms these reactions are added automatically if the inhibitor/activator and metabolites affected by the inhibitor are specified.

Finally, once the elementary reactions are specified, a mass action rate law is generated for each elementary reaction.

For ENO, we decompose the overall reaction

$$2PG \rightleftharpoons PEP$$

into the following elementary steps:

$$ENO + 2PG \rightleftharpoons ENO\&2PG \rightleftharpoons ENO\&PEP \rightleftharpoons ENO + PEP$$

where $ENO\&2PG$ is the complex formed by the enzyme and the substrate 2PG. In this case there is only one catalytic pathway, which contains all the elementary reactions defined above. We then generate a mass action rate law for each step:

$$\begin{aligned} v_1 &= \overrightarrow{k}_1[ENO][2PG] - \overleftarrow{k}_1[ENO\&2PG] \\ v_2 &= \overrightarrow{k}_2[ENO\&2PG] - \overleftarrow{k}_2[ENO\&PEP] \\ v_3 &= \overrightarrow{k}_3[ENO\&PEP] - \overleftarrow{k}_3[ENO][PEP] \end{aligned}$$

where the square brackets denote the time dependent concentration of the enzyme form/metabolite. By enzyme forms we mean both free enzyme and enzyme bound to substrate(s)/product(s).

If ENO is activated by metabolite A, we would add the following set of elementary reactions:

$$\begin{aligned} &ENO + A \rightleftharpoons ENO\&A \\ &ENO\&A + 2PG \rightleftharpoons ENO\&A\&2PG \\ &ENO\&A\&2PG \rightleftharpoons ENO\&A\&PEP \\ &ENO\&A\&PEP \rightleftharpoons ENO\&A + PEP \end{aligned}$$

If, instead, ENO was inhibited competitively by metabolite I with regards to $2PG$ we would add the elementary reaction:

$$ENO + I \rightleftharpoons ENO\&I$$

To illustrate the importance of defining catalytic pathways, we will consider fructose-bisphosphate aldolase class I (FBA1) and assume the products are released in random order for illustrative purposes. FBA1 catalyzes the reaction:

$$FDP \rightleftharpoons DHAP + G3P$$

the elementary reactions would be:

$$\begin{aligned}
&FBA1 + FDP \rightleftharpoons FBA1\&FDP\\
&FBA1\&FDP \rightleftharpoons FBA1\&DHAP\&G3P\\
&FBA1\&DHAP\&G3P \rightleftharpoons FBA1\&DHAP + G3P\\
&FBA1\&DHAP \rightleftharpoons FBA1 + DHAP\\
&FBA1\&DHAP\&G3P \rightleftharpoons FBA1\&G3P + DHAP\\
&FBA1\&G3P \rightleftharpoons FBA1 + G3P
\end{aligned}$$

This mechanism leads to two individual catalytic pathways:

$$\begin{aligned}
1:&FBA1 + FDP \rightleftharpoons FBA1\&FDP \rightleftharpoons FBA1\&DHAP\&G3P \rightleftharpoons\\
&\rightleftharpoons FBA1\&DHAP + G3P \rightleftharpoons FBA1 + DHAP
\end{aligned}$$

$$\begin{aligned}
2:&FBA1 + FDP \rightleftharpoons FBA1\&FDP \rightleftharpoons FBA1\&DHAP\&G3P \rightleftharpoons\\
&\rightleftharpoons FBA1\&G3P + DHAP \rightleftharpoons FBA1 + G3P
\end{aligned}$$

where each track needs to satisfy the respective Haldane relation, i.e.,

$$\begin{aligned}
\frac{\prod_{i=1}^{n} \overrightarrow{k_{1i}}}{\prod_{i=1}^{n} \overleftarrow{k_{1i}}} &= K_{eq,FBA1}\\
\frac{\prod_{i=1}^{n} \overrightarrow{k_{2i}}}{\prod_{i=1}^{n} \overleftarrow{k_{2i}}} &= K_{eq,FBA1}
\end{aligned}$$

where $\overrightarrow{k_{1i}}$ and $\overleftarrow{k_{1i}}$ represent the rate constants in the forward and reverse directions, respectively, for the catalytic pathway one, and $\overrightarrow{k_{2i}}$ and $\overleftarrow{k_{2i}}$ represent the rate constants for the catalytic pathway two. n is the number of elementary reactions in the respective catalytic pathway. $K_{eq,FBA1}$ is the equilibrium constant for FBA1.

Constructing symbolic comparison equations

Once kinetic data are processed, equations that represent the model behavior equivalent to these data types are generated to begin setting up a non-linear least squares problem. These equations are specific to each data type. For K_{eq} values, Haldane relationships are defined as the ratio of forward and reverse rate constants for each catalytic pathway. For ENO, the Haldane relation used to fit the reaction's K_{eq} is:

$$K_{eq} = \frac{\overrightarrow{k}_1 \overrightarrow{k}_2 \overrightarrow{k}_3}{\overleftarrow{k}_1 \overleftarrow{k}_2 \overleftarrow{k}_3}$$

For k_{cat} and K_m values, equations are derived from the overall steady-state flux equation, v_{ss}. The reason we use steady-state equations to fit the data is because we are fitting data which was determined by assuming the steady-state. The overall flux equation for each reaction is assumed to be the same as the flux through the conversion elementary reaction, i.e. the elementary reaction in which substrates are converted into products. By solving the system of mass balance equations at steady-state for all enzyme-form concentrations together with a total enzyme sum equation, we find the enzyme-form concentrations at steady-state in terms of rate constants $\boldsymbol{k}$ and metabolite concentrations. By substituting the steady-state concentrations of the enzyme-substrate(s) and enzyme-product(s) complexes into the overall flux equation, we get the overall steady-state equation. This overall steady-state equations is now defined in terms of rate constants, total enzyme concentration, and metabolite concentrations.

For ENO, the overall flux equation is:

$$v_{ENO} = v_2 = \overrightarrow{k}_2[ENO\&2PG] - \overleftarrow{k}_2[ENO\&PEP] \tag{2.1}$$

To determine the overall steady-state flux equation, we need to find the steady-state concentration for $[ENO\&2PG]$ and $[ENO\&PEP]$ in terms of rate constants $\boldsymbol{k}$ and metabolite concentrations $[2PG]$ and $[PEP]$. To do so, we solve the following

system of steady-state equations for all enzyme-form concentrations together with the total enzyme conservation equation:

$$\begin{aligned}
&\frac{d}{dt}[ENO] = v_3 - v_1 = 0 \\
&\frac{d}{dt}[ENO\&2PG] = v_1 - v_2 = 0 \\
&\frac{d}{dt}[ENO\&PEP] = v_2 - v_3 = 0 \\
&ENO_{tot} = [ENO] + [ENO\&2PG] + [ENO\&PEP]
\end{aligned} \tag{2.2}$$

After solving this system of equations we know

$$[ENO\&2PG] = h_{[ENO\&2PG]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$$

and

$$[ENO\&PEP] = h_{[ENO\&PEP]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$$

at steady-state, and substitute these into equation system 2.2 to get:

$$v_{ss} = f_{v_{ss}}(ENO, 2PG, PEP, \boldsymbol{k}) \tag{2.3}$$

$$= \overrightarrow{k}_2 \cdot h_{[ENO\&2pg]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k}) - \overleftarrow{k}_2 \cdot h_{[ENO\&pep]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$$

$$= \frac{[ENO_{tot}]([2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3)}{[2PG]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3 + [PEP]\overleftarrow{k}_3(\overleftarrow{k}_1 + \overleftarrow{k}_2 + \overrightarrow{k}_2)}$$

This overall steady-state flux equation becomes an effective k_{cat} value, $k_{cat,eff}$, when divided by an arbitrary total enzyme value, which we set to 1, and given saturating concentrations of substrates and no product concentrations. It can then be used for direct comparison to k_{cat} ($k_{cat} = V_{max}/[E_{tot}]$). The motivation to use an effective k_{cat} equation instead of the theoretical equation for k_{cat} – which is based on the limit of the steady-state flux equation when product concentration is zero and substrate concentration tends to infinity – is two-fold: 1) to the best of our knowledge, k_{cat} values are typically determined by using saturating substrate concentrations and

fitting the reaction's rate to the irreversible Michaelis-Menten equations for a one substrate reaction, using the Lineweaver-Burk plot [25] or similar methods. In that case, substrate concentrations may be saturating, but not infinite. Thus, the measured k_{cat} is actually an effective k_{cat} and not the theoretical one. Furthermore, k_{cat} values are often reported for different substrate concentrations, which we would not be able to fit with a theoretical k_{cat} equation; 2) for reactions with more than one substrate taking the limit of the forward flux equation when all substrates concentrations tend to infinity is not a straightforward task.

For ENO, to generate the equation used to fit the forward k_{cat} value, we assume $v_{ss} = V_{max,eff}$ and $k_{cat,eff} = V_{max,eff}/[E_{tot}]$ when substrate concentrations are saturating and product concentrations are null. In practice, this means that we divide v_{ss} by $[ENO_{tot}]$ and set $[PEP] = 0$, resulting in:

$$k_{cat,eff} = \frac{[2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3}{[2PG]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3} \tag{2.4}$$

By using an effective k_{cat} equation to fit the measured k_{cat} values, we can fit different k_{cat} values measured for different substrate concentrations.

To generate the equation to fit K_m values, equations for the relative velocity of the enzyme compared to $V_{max,eff}$ ($v/V_{max,eff}$) are defined by dividing the overall rate law by the same rate law when the substrate concentration is saturating. In general we assume saturating concentrations to be 1 M, but the value can be changed by the user.

For ENO, to generate the equations used to fit the effective K_m for $2PG$, we take equation 2.4 and divide it by itself when $[2PG]$ is saturating, i.e. 1 M, leading to:

$$v_{rel} = \frac{[2PG](\overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2)) + \overrightarrow{k}_2\overrightarrow{k}_3 + \overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2)}{[2PG]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3}$$

This equation can take values from 0 to 1, and by solving it for $[2PG]$ when $v_{rel} = 0.5$ we get the effective K_m value for 2PG.

To fit inhibition constants, we also use a relative rate flux equation with the appropriate metabolite concentrations defined. If the inhibition step is a dead-end, i.e. once the inhibitor binds the reaction cannot progress, we also fit the inhibition constant to the inhibitor dissociation constant.

Preparing kinetic data for fitting

The kinetic data, such as K_{eq}, k_{cat}, K_m, and K_i, are then converted to a form that allows direct comparison to model behavior. For K_{eq} values, the form is simply the value itself as well as the corresponding conditions (pH, IS, and T). For k_{cat}, the form is again the value as well as the measured experimental conditions, including substrate concentrations. If substrate concentrations are not available, a concentration that is likely to be saturating (e.g.. 1M), is assumed, to represent the excess concentrations typically used in the measurement of turnover rates. For K_m and K_i, an initial rate curve is generated using the classical Michaelis-Menten equation with the K_m and K_i value substituted when applicable. This curve is simulated at different substrate concentrations in between an order of magnitude above and below the measured K_m value. Other experimental conditions such as co-substrate concentration and media conditions are reported when available. In case we are fitting a K_i this curve is generated for different inhibitor concentrations. If instead of a Michalis-Menten constant we have a $S_{0.5}$ value and the respective Hill coefficient, a Hill equation is used to generate the data to be fitted. This procedure attempts to simulate the original experiment. In principle, instead of this experimental plot simulation procedure, raw data could be used as well. Furthermore, by simulating the initial rate curve, we can fit different types of kinetics, e.g. Michaelis-Menten kinetics or Hill kinetics, as long as the enzyme mechanism is specified correctly.

To attempt to correct these values for *in vitro* to *in vivo* differences, we can apply a couple of adjustments to the data. First, k_{cat} can be adjusted to *in vivo* temperature from *in vitro* conditions using a user defined Q_{10} temperature coefficient value for the enzyme. The Q_{10} coefficient defines, in this case, how much the k_{cat} value increases as a response to a temperature increase of 10 C. The Q_{10} coefficient is initially assumed to have a typical value of 2.5. However, the Q_{10} for mesophilic enzymes can range from ~ 1.5 to ~ 3 [11, 31, 43]. Second, inhibitory effects of pH changes on enzyme behavior can be modeled by adding proton binding and dissociation reactions to the enzyme mechanism with associated dissociation constants, as has been done in the enzymology literature [40].

For ENO, the data table to be generated is the following:

Weight	$[2PG]$	$[PEP]$	$[ENO_{tot}]$	pH	T (°C)	Function file	Target value
1	0	0	1	7.5	25	haldaneRatio_1.txt	5.19
1	0.01	0	1	8.1	30	absRateFor.txt	330
1	0.00001	0	1	8.1	30	relRateFor_2pg_1.txt	0.091
1	0.000016	0	1	8.1	30	relRateFor_2pg_1.txt	0.137
1	0.000025	0	1	8.1	30	relRateFor_2pg_1.txt	0.201
1	0.00004	0	1	8.1	30	relRateFor_2pg_1.txt	0.285
1	0.000063	0	1	8.1	30	relRateFor_2pg_1.txt	0.387
1	0.0001	0	1	8.1	30	relRateFor_2pg_1.txt	0.5
1	0.00016	0	1	8.1	30	relRateFor_2pg_1.txt	0.613
1	0.00025	0	1	8.1	30	relRateFor_2pg_1.txt	0.715
1	0.0004	0	1	8.1	30	relRateFor_2pg_1.txt	0.799
1	0.00063	0	1	8.1	30	relRateFor_2pg_1.txt	0.863
1	0.001	0	1	8.1	30	relRateFor_2pg_1.txt	0.909

where the first entry is the data point to fit the K_{eq}, the second entry is the data point to fit the k_{cat}, and all the other data points are used to fit the K_m for $2PG$. One way to address the uncertainty associated with the measurement of kinetic data is to generate different datasets for different data values sampled from the uncertainty interval defined in the table 2.1. These datasets would then be fitted independently.

Two-stage randomized fitting

Once both the data and equations have been prepared, these are passed to a two-stage nonlinear least squares optimization procedure. The target values, i.e. the values to be fitted by the equations, are given by the data, while the model values are given by the equations with the experimental conditions substituted into them . Once the experimental conditions are substituted into the equations, these become a function of the rate constants only. The fitting procedure then yields rate constants with which the enzyme model can reproduce the measured data. As an objective, we use the absolute difference between the logarithm of the predicted data values and the logarithm of the data points being fitted. This difference is then multiplied by the user-defined weights for each data point. The logarithm is used here so that we can fit parameters with different orders of magnitude. We use bounds on rate constants of 10^{-6} s^{-1} and 10^{9} s^{-1} based on generally assumed limits of diffusion and

an arbitrarily slow lower bound [2]. These bounds may not be relevant for reactions not involving association or dissociation, such as conversion steps. However, we have not found these bounds to affect the error for the cases we have examined thus far. Additionally, to address the poor scaling of rate constants, we log transform the rate constants during this optimization.

Yet, the equations are highly nonlinear, causing many optimization algorithms to fail to converge to an acceptable fit. To address this challenge, we first run a non-derivative based particle swarm optimization (PSO) to find rate constant sets that fit the data well enough to serve as initial points for a more precise derivative-based optimization. The second optimization is a Levenberg-Marquadt (LMA) derivative-based optimization with the same log-parameter scaling and objective. Resulting fits are examined for total residual and points that fit satisfactorily are kept. These rate constants are then once more substituted into the equations for the kinetic data to verify that the enzyme kinetic data such as K_m, k_{cat}, and K_{eq} are reproduced. Another way to address the uncertainty associated with the measurement of kinetic data, is to consider fitted rate constant sets that do not exactly reproduce the kinetic data but do so within 5-10% of the measured value.

Importantly, as PSO is a randomized algorithm, initial points passed to LMA are different each time the algorithm is run. As a result, different rate constant sets are returned by the optimization every time, each of which fits the kinetic data equivalently well. This effectively samples the rate constant space in the prevalent cases where rate constants are not uniquely specified by the available kinetic data. Thus, this procedure inherently addresses the under-specification issue in parameterization of microscopic enzyme reaction mechanisms.

Cluster parameters

The fitting procedure is then repeated a number of times until the space of equivalent optima has been sufficiently sampled. As initial points to the LMA optimization are defined pseudo-randomly based on the PSO optimization results, the resulting rate constants are usually different in different optimizations. To check whether the space of equivalent optima has been sufficiently sampled we take each resulting rate constant set, and discretize the rate constant values by binning them. Since $k \in [10^{-6}, 10^{9}]$, we consider the following bins $[10^{-6}, 10^{-3}[$, $[10^{-3}, 10^{0}[$, $[10^{0}, 10^{3}[$, $[10^{3}, 10^{6}[$, $[10^{6}, 10^{9}]$. We call a pattern to each binned set of rate constants. For instance, if we have the set of rate constants (2.1×10^{3}, 1.5×10^{8}, 7.8×10^{1}, 4.2×10^{3}, 6.4×10^{4}, 5.1×10^{8}), this would lead to the following pattern (4, 5, 3, 4, 4, 5), where each number corresponds

to a bin. If new fitted rate constant sets fall mostly into existing patterns, we consider the sampling space to have been sufficiently covered.

Integrate omics data to build fully functional enzyme-level models (optional)

Once rate constant sets are determined, we use steady-state metabolomics data to define the initial metabolite concentrations. To determine all enzyme form initial concentrations, we can either use steady-state fluxomics or proteomics. Once the reaction's rate constants, initial metabolite concentrations, and initial enzyme form concentrations are specified the model is ready to be simulated.

To determine the enzyme form initial concentrations for ENO using steady-state proteomics data, we take the system of steady-state equations for all enzyme forms generated earlier:

$$\begin{aligned}
\frac{d}{dt}[ENO] &= v_3 - v_1 = 0 \\
\frac{d}{dt}[ENO\&2PG] &= v_1 - v_2 = 0 \\
\frac{d}{dt}[ENO\&PEP] &= v_2 - v_3 = 0
\end{aligned}$$

solving this system of equations for the enzyme form concentrations, leads to:

$$\begin{aligned}
[ENO] &= \frac{[ENO_{tot}]\,A}{[2PG]\,B + [PEP]\,C + D} \\
[ENO\&2PG] &= \frac{[ENO_{tot}]\,A}{[2PG]\,B + [PEP]\,C + D} \\
[ENO\&PEP] &= \frac{[ENO_{tot}]([2PG]\,E + [PEP]\,F)}{[2PG]\,B + [PEP]\,C + D}
\end{aligned} \tag{2.5}$$

where

$$
\begin{aligned}
A &= \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3 \\
B &= \overrightarrow{k}_1\overrightarrow{k}_3 + \overrightarrow{k}_1\overleftarrow{k}_2 + \overrightarrow{k}_1\overrightarrow{k}_2 \\
C &= \overleftarrow{k}_3(\overleftarrow{k}_1 + \overleftarrow{k}_2 + \overrightarrow{k}_2) \\
D &= \overrightarrow{k}_2\overrightarrow{k}_3 + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) \\
E &= \overrightarrow{k}_1\overrightarrow{k}_2 \\
F &= \overleftarrow{k}_3(\overleftarrow{k}_1 + \overrightarrow{k}_2)
\end{aligned}
$$

By substituting the rate constant values, and the steady-state concentrations for ENO, 2PG, and PEP, we obtain the concentrations for all the enzyme forms: free enzyme and enzyme bound to 2PG and PEP.

If we want to determine the enzyme form concentrations by integrating steady-state flux data instead of proteomics, we can solve the steady-state flux equation 2.3 for $[ENO_{tot}]$ and then substitute $[ENO_{tot}]$ in equation 2.5. Thus, taking the original steady-state flux equation,

$$
v_{ss} = \frac{[ENO_{tot}]([2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3)}{[2PG]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3 + [PEP]\overleftarrow{k}_3(\overleftarrow{k}_1 + \overleftarrow{k}_2 + \overrightarrow{k}_2)}
$$

we solve it for $[ENO_{tot}]$ and get:

$$
[ENO_{tot}] = \frac{v_{ss}([PEP]\,A + [2PG]\,B + C)}{[2PG]\,D - [PEP]\,E}
$$

where

$$
\begin{aligned}
A &= \overleftarrow{k}_1\overleftarrow{k}_3 + \overleftarrow{k}_2\overleftarrow{k}_3 + \overleftarrow{k}_3\overrightarrow{k}_2 \\
B &= \overrightarrow{k}_1\overrightarrow{k}_3 + \overrightarrow{k}_1\overrightarrow{k}_2 + \overrightarrow{k}_1\overleftarrow{k}_2 \\
C &= \overleftarrow{k}_1\overrightarrow{k}_3 + \overleftarrow{k}_1\overleftarrow{k}_2 + \overrightarrow{k}_2\overrightarrow{k}_3 \\
D &= \overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 \\
E &= \overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3
\end{aligned}
$$

this leads to:

$$[ENO] = \frac{v_{ss}(\overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3)}{[2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3}$$

$$[ENO\&2PG] = \frac{v_{ss}([PEP]\overleftarrow{k}_2\overleftarrow{k}_3 + [2PG]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2))}{[2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3}$$

$$[ENO\&PEP] = \frac{v_{ss}([2PG]\overrightarrow{k}_1\overrightarrow{k}_2 + [PEP]\overleftarrow{k}_3(\overleftarrow{k}_1 + \overrightarrow{k}_2)}{[2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3}$$

where we simply need to substitute [PEP] and [2PG] for their steady-state concentrations and v_{ss} for the steady-state flux for ENO.

Now that we know all the initial metabolite and enzyme concentrations are known, we can simulate our enzyme-level ODE model:

$$\begin{aligned}
\frac{d}{dt}[2PG] &= \overleftarrow{k}_1[ENO\&2PG] - \overrightarrow{k}_1[ENO][2PG] \\
\frac{d}{dt}[PEP] &= \overrightarrow{k}_3[ENO\&PEP] - \overleftarrow{k}_3[ENO][PEP] \\
\frac{d}{dt}[ENO] &= \overleftarrow{k}_1[ENO\&2PG] - \overrightarrow{k}_1[ENO][2PG] + \overrightarrow{k}_3[ENO\&PEP] \\
&\quad - \overleftarrow{k}_3[ENO][PEP] \\
\frac{d}{dt}[ENO\&2PG] &= -\overrightarrow{k}_2[ENO\&2PG] + \overleftarrow{k}_2[ENO\&PEP] - \overleftarrow{k}_1[ENO\&2PG] \\
&\quad + \overrightarrow{k}_1[ENO][2PG] \\
\frac{d}{dt}[ENO\&PEP] &= \overrightarrow{k}_2[ENO\&2PG] - \overleftarrow{k}_2[ENO\&PEP] - \overrightarrow{k}_3[ENO\&PEP] \\
&\quad + \overleftarrow{k}_3[ENO][PEP]
\end{aligned}$$

2.2.1 Case studies

Having described the workflow for parameterizing rate constants for elementary mass action reaction mechanisms using enzyme kinetic data, we now present four case studies demonstrating the workflow in action when different types of data are available. The Mathematica® notebooks for these case studies are available as examples in the

github repository (https://github.com/martamatos/MASSef). For all case studies the k_{cat} values were corrected to 37 C using a Q_{10} of 2.5. The K_{eq} was obtained from the eQuilibrator website [13] for pH 7.5 and IS 0.25 M. The assumed total enzyme concentration was 1 M. We used scipy's [22] hierarchical clustering (`hierarchy()` function) to cluster all rate constant sets, we chose 'Ward' as the clustering method and the euclidean distance as the distance metric.

Case study 1: Fitting the classical enzyme kinetic constants k_{cat}, K_m, and K_{eq} for Triose-phosphate Isomerase (TPI)

We start by using MASSef to fit a simple enzyme, TPI, which converts D-glyceraldehyde 3-phosphate (G3P) into dihydroxyacetone phosphate (DHAP). The enzymatic mechanism is depicted in Fig. 2.2A, and the available kinetic data for TPI is shown in Fig. 2.2B. All data point weights were set to 1. We fit the enzyme 100 times, so that we generate 100 rate constant sets. Only rate constant sets that lead to a sum of squared log-deviations (ssld) less than 0.1 are considered. The sum of squared log-deviations is defined as:

$$ssld = \sum_{i=0}^{n} (\log(x_{data_i}) - \log(x_{predicted_i}))^2$$

Where $x_{predicted}$ are the values that result from substituting the estimated rate constant values into the equations used for fitting the kinetic data. The reason why we take the logarithm of the data points to be fitted and the predicted data points, is that the data points used to fit the data have different orders of magnitude. Thus, by taking the logarithm, when we select a given threshold for the ssld, we get rate constant sets that fit all data points similarly well, instead of getting rate constant sets that fit one data point very well, but do not fit another data point at all.

For TPI, an ssld below 0.1 means that the highest deviation from the true value is lower than 0.06%. All 100 fits have an ssld below 0.1.

The equations used for fitting are the following:

$$k_{cat} = \frac{[G3P]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3}{[G3P]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3}$$

To generate the equation used to fit the K_m for $G3P$, we take the above equation and divide it by itself when $[G3P] \to 1$ M, i.e. the G3P concentration is saturating, leading to:

$$\frac{v_{ss}}{V_{max,eff}} = \frac{[G3P]\overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3 + \overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2)}{[G3P]\overrightarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2 + \overrightarrow{k}_2) + \overleftarrow{k}_1(\overrightarrow{k}_3 + \overleftarrow{k}_2) + \overrightarrow{k}_2\overrightarrow{k}_3}$$

When we solve $v_{ss}/V_{max,eff} = 0.5$ for [G3P], the result is the effective K_m for G3P. Finally, the Haldane relation used to fit the reaction's equilibrium constant is:

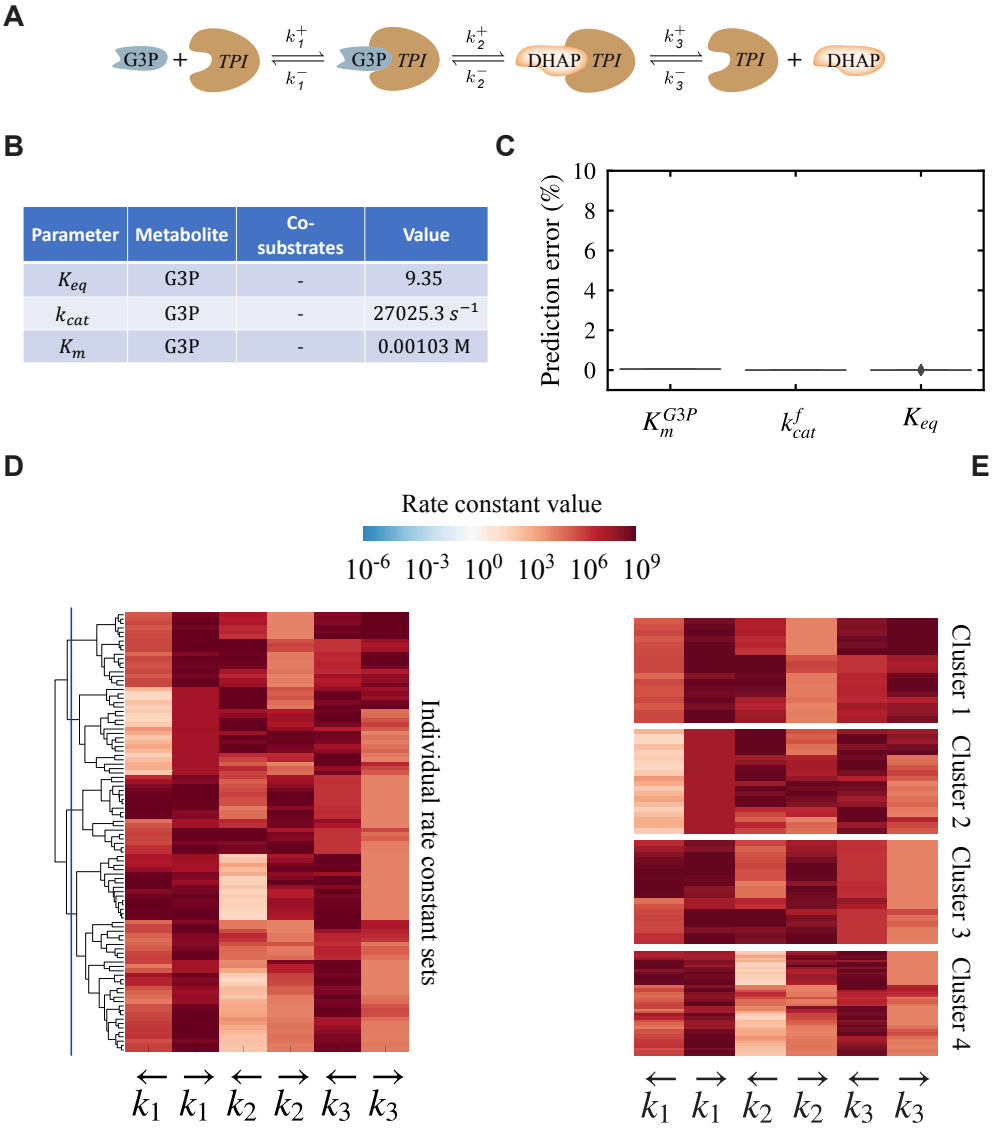
$$K_{eq} = \frac{\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3}{\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3}$$

These equations are also used to obtain the kinetic parameters predicted values after fitting the enzyme, i.e. for each rate constant set, we integrate into the above equations the fitted rate constant values and the metabolite/enzyme concentrations used for the fitting. In Fig. 2.2C, we show the prediction error for each kinetic parameter, calculated as

$$\text{Prediction error}\,(\%) = \frac{x_{data} - x_{predicted}}{x_{data}} \times 100$$

where x is the respective kinetic parameter, K_m, k_{cat}, or K_{eq}. Overall the error is very low.

In Fig. 2.2D we show the clustering of all rate constant sets, where the rate constant values range from 10^0 to 10^9. Overall, the forward rate constant for the binding step, $\overrightarrow{k_1}$, is faster than the respective reverse rate constant, $\overleftarrow{k_1}$, while the opposite is true for the product release step. Setting the cut-off distance between clusters to 5, we distinguish four clusters in Fig. 2.2E. Here, we can see that in the first cluster the forward rate constant for the conversion step, $\overrightarrow{k_2}$, tends to be lower than the respective reverse rate constant, $\overleftarrow{k_2}$, while $\overrightarrow{k_3}$ tends to be higher than $\overleftarrow{k_3}$, both being generally high. On cluster three, however, $\overrightarrow{k_3}$ is generally lower than $\overleftarrow{k_3}$ and $\overrightarrow{k_2}$ is generally higher than $\overleftarrow{k_2}$. The same is true for cluster four, however the magnitudes for $\overrightarrow{k_2}$ and $\overleftarrow{k_2}$ are lower than in cluster three and $\overleftarrow{k_3}$ has higher magnitude than in cluster three. This seems to indicate that, as expected, the rate constant values are not independent, i.e. if the value of a given rate constant changes then another rate constant value(s) need to change as well to fit the kinetic data properly.



Parameter	Metabolite	Co-substrates	Value
K_{eq}	G3P	-	9.35
k_{cat}	G3P	-	27025.3 s^{-1}
K_m	G3P	-	0.00103 M

Figure 2.2: (A) The enzyme mechanism for TPI. (B) The kinetic data used to fit the enzyme. (C) The relative error associated with the prediction of each parameter from each rate constant set. (D) Clustering of all rate constant sets. The blue bar indicates the cutoff distance to define individual clusters. (E) Clusters obtained when setting the cutoff distance to 5.

To assess the reproducibility of the above results we repeated the set of 100 fits three times. The results are very similar between the three resulting sets of 100 fits. To try and quantify how similar the rate constant sets are between different sets of 100 fits, we discretize their values by binning them and generate a pattern for each rate constant set, as explained previously.

For each of the three sets of 100 fits, we bin all the rate constant sets and find that all three sets of 100 fits have 15 patterns in common, and that each set of 100

fits has between 3 and 4 patterns that are not present in any of the other two sets. . This seems to indicate that for TPI 100 fits are enough to explore the solution space, and that the amount of combinations of rate constant values that fit the kinetic data well is relatively small.

Case study 2: Handling parameters measured under different conditions in Fructose-bisphosphate Aldolase class I (FBA1)

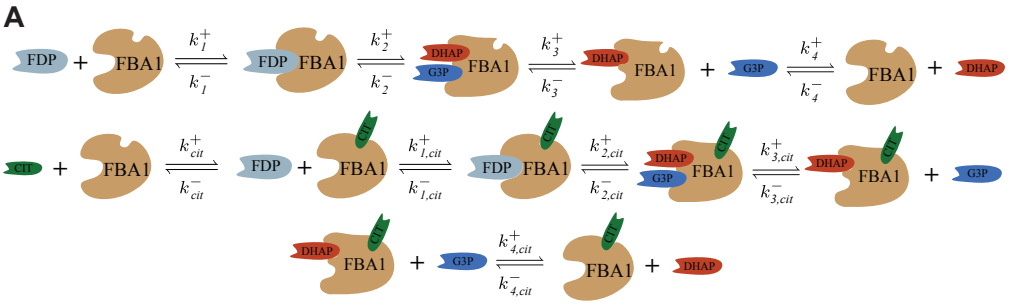
With FBA1, we show MASSef's ability to handle measurements of the same kinetic parameter in different conditions. For FBA1 we have k_{cat} and K_m values measured both in the presence and absence of citrate, see Fig. 2.3B. To be able to fit both measurements we consider two sets of elementary reactions, i.e. two catalytic pathways: one in which citrate is bound to FBA1 and one in which it is not, as depicted in Fig. 2.3A. To increase the number of good fits, we set the weights for K_m to 10, the k_{cat} measured in the absence of citrate and the K_{eq} to 2, while the weight for the k_{cat} measured in the presence of citrate is set to 1. For each 100 enzyme fits we only get 5-6 fits with an ssld below 0.005, which in this case means that the error associated to the kinetic parameters prediction is less than 26%. Thus, to get more good fits and analyze the results, we fit the enzyme 1000 times and get 57 fits with an ssd below 0.005. Most likely, the reason it is harder to fit this enzyme, in the sense that most of the sampled rate constant sets do not fit the kinetic data, is that we are fitting different values for the same parameters.

The results for the rate constant sets that lead to an ssld below 0.005 are shown in Figs. 2.3C-D. In Fig. 2.3C we see that, overall, the error associated to the prediction of each data point is very small, with some outliers for the equilibrium constants, where the highest error is $\sim 5\%$ for the K_{eq} associated to the citrate catalytic pathway. In Fig. 2.3D we show the clustering results of all rate constant sets with ssd below 0.005. Overall the rate constant sets are quite different from each other, especially when comparing to TPI. Since, in this case, it is hard to interpret the clustering results in terms of rate constants, we convert each set of rate constants into a set of elementary equilibrium constants, where an elementary equilibrium constant is defined as

$$K_{eq,i} = \frac{\overrightarrow{k_i}}{\overleftarrow{k_i}}$$

In Fig. 2.4A we show the clustering results in terms of elementary equilibrium constants, and by using a distance cutoff of 33, we distinguish three different clusters in Fig. 2.4B. In cluster one the release of DHAP in the absence of citrate, K_4, seems to be slightly more favorable than in the other clusters, while the release of G3P both in the absence and presence of citrate, K_3 and $K_{3,cit}$, seems less favorable than in the other clusters. In cluster three the release of G3P in the presence of citrate, $K_{3,cit}$, seems more favorable than in clusters one and two. Overall the binding of citrate, K_{cit}, seems to be favorable in all clusters, as well as the binding of FBP, K_1 and $K_{1,cit}$. While the conversion step, K_2 and $K_{2,cit}$, seems to be unfavorable in all clusters

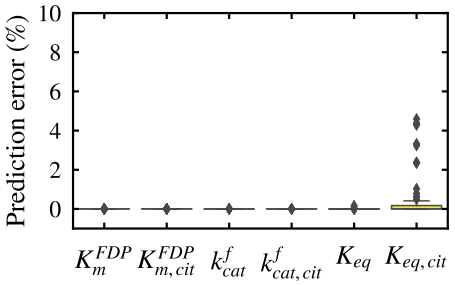
A



B

Parameter	Metabolite	Co-substrates	Value
K_{eq}	FDP	-	0.00016
k_{cat}	FDP	-	0.41 s^{-1}
k_{cat}	FDP	citrate, 0.011 M	6 s^{-1}
K_m	FDP	-	0.00002 M
K_m	FDP	citrate, 0.011 M	0.00007 M

C



D

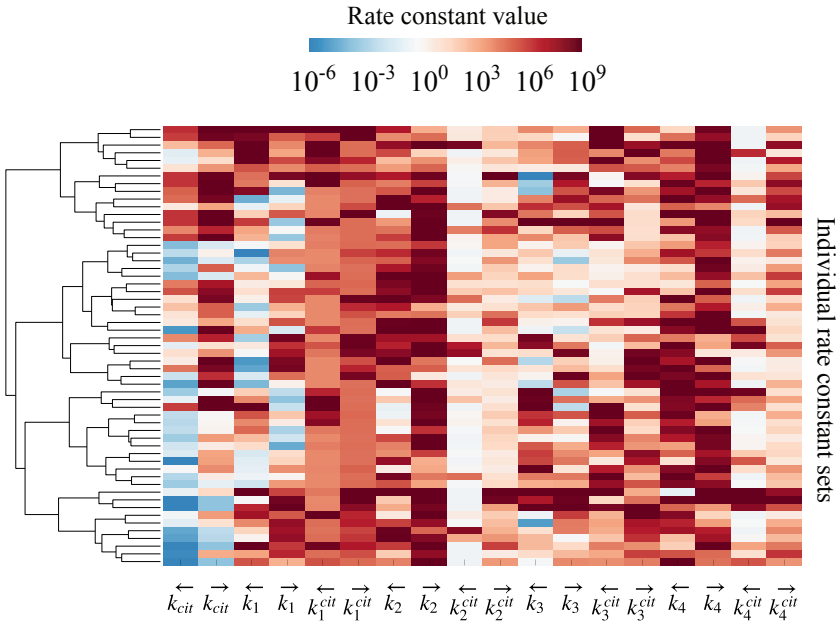


Figure 2.3: (A) Enzyme mechanism for FBA1 with two catalytic pathways, one where citrate is absent and the other where citrate is bound to the enzyme. (B) Data used to fit the enzyme rate constants. (C) Error associated to the prediction of each data point used in the fitting. (D) Clustering of rate constant sets with ssld lower than 0.005.

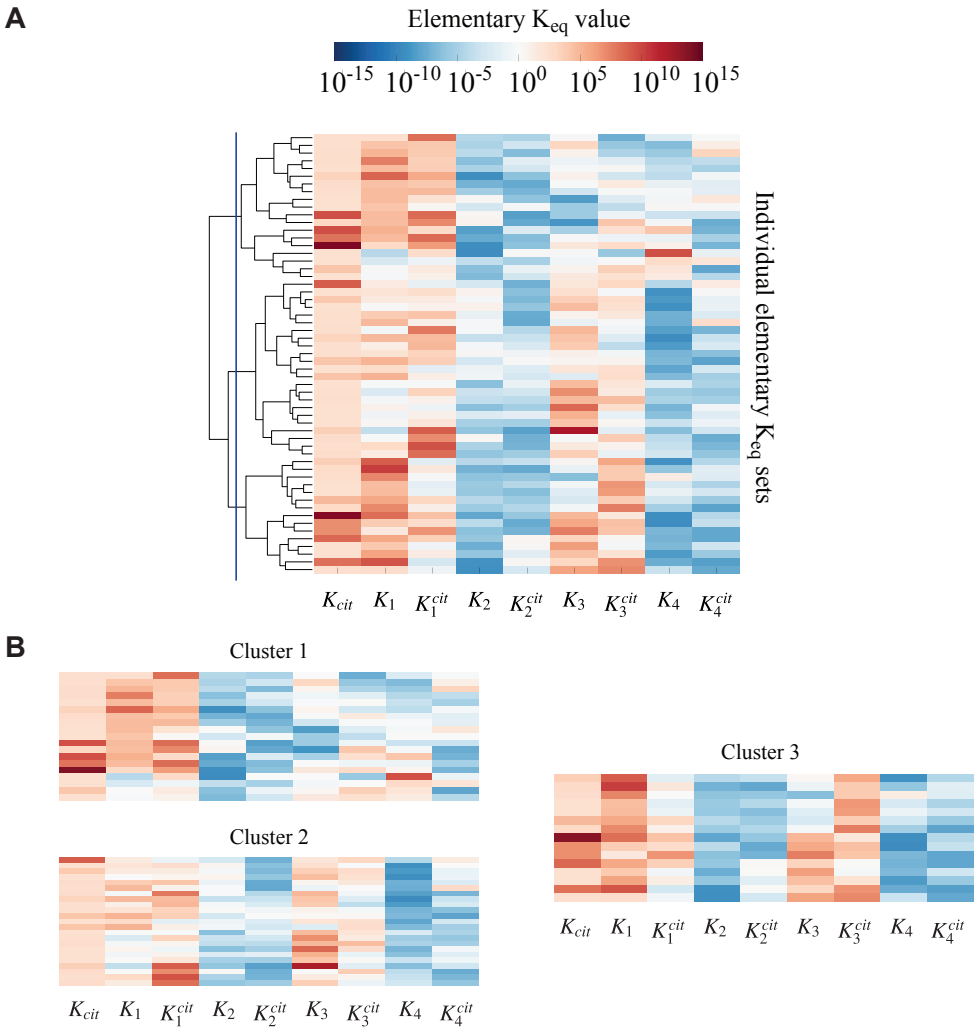


Figure 2.4: (A) Clustering of all sets of elementary equilibrium constants with ssld< 0.005. The blue bar indicates the cutoff distance to define individual clusters. (B) Clusters obtained when the cutoff distance between clusters is set to 33.

To fit the equilibrium constant we have two Haldane relations, one for the catalytic pathway where citrate is bound to the enzyme and the other where citrate is not bound to the enzyme:

$$K_{eq} = \frac{\overrightarrow{k}_1 \overrightarrow{k}_2 \overrightarrow{k}_3 \overrightarrow{k}_4}{\overleftarrow{k}_1 \overleftarrow{k}_2 \overleftarrow{k}_3 \overleftarrow{k}_4}$$

$$K_{eq,cit} = \frac{\overrightarrow{k}_{1,cit} \overrightarrow{k}_{2,cit} \overrightarrow{k}_{3,cit} \overrightarrow{k}_{4,cit}}{\overleftarrow{k}_{1,cit} \overleftarrow{k}_{2,cit} \overleftarrow{k}_{3,cit} \overleftarrow{k}_{4,cit}}$$

The equations used to fit the effective k_{cat} and K_m for FBP can be found in section 2.4.2.1.

To assess the results reproducibility in terms of rate constant sets, we repeated the set of 1000 fits three times. We binned the rate constant sets, as we did for TPI, and checked how many resulting patterns were common among the three sets of 1000 fits. We considered only rate constant sets with an associated ssld below 0.005. We found no common patterns among the three sets of fits. This might be due to the high number of rate constants, which means that many more combinations are possible. Also, there are more combinations of rate constant sets that can fit the kinetic data equally well, and 1000 fits are apparently not enough to find all the possible combinations.

However, when we use these rate constant sets to build a full ODE model for the enzyme and simulate the resulting models as a closed system, the metabolite time-courses are fairly similar, see Fig. 2.5. Each column in the figure represents the resulting enzyme-level ODE model ensemble for the respective set of 1000 fits, considering only the rate constant sets where ssld < 0.005. The omics data used to generate the model can be found in section 2.4.8.

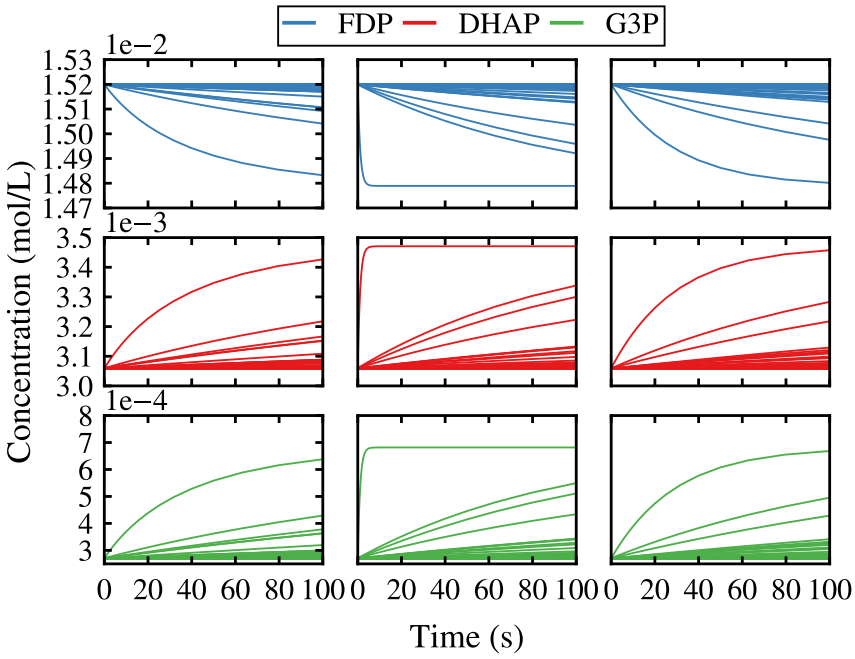


Figure 2.5: Time-course metabolite concentrations for three enzyme-level model ensembles for FBA1. Each row corresponds to a different metabolite, from top to bottom: FDP, DHAP, and G3P. Each column corresponds to a different model ensemble, each with 59 models. Each line corresponds to one model.

Case study 3: Handling sigmoidal kinetics in Fructose-1,6-bisphosphatase 2 (FBP2)

Here, we show the ability of MASSef to fit not only Michaelis-Menten kinetics, but also Hill kinetics. We use FBP2, an enzyme with two subunits and two active sites showing Hill kinetics with a Hill coefficient of 2.05. Even though more kinetic data is available for this enzyme, we only use the data shown in Fig. 2.6B. This is because we want to focus on the sigmoidal kinetic behavior. The enzymatic mechanism assumed is depicted in Fig. 2.6A, where we assume, as a simplification, that both molecules of F6P are released before any phosphate molecule is released. The results in Figs. 2.6C-E include only rate constant sets with ssld lower than 0.1, which, in this case means that the errors associated to the kinetic parameters prediction are below 5%. We got 90 rate constant sets out of 100 with ssld lower than 0.1. All data point priorities were set to 1.

In Fig. 2.6C we show the errors associated with the prediction of each data point, which are almost null for K_{eq} and k_{cat}, but just below 4% for $S_{0.5}^{FBP}$ and slightly above 2% for the Hill coefficient. This means that the sigmoidal curve is

not reproduced perfectly, but the deviation from the true curve is most likely still within the uncertainty of the experimental measurements. In Fig. 2.6D we show the clustering results of all elementary equilibrium constant sets with ssld below 0.1. We do not show the rate constant sets because we have again 14 rate constants which become hard to interpret, as in the FBA1 case study. Overall the FBP binding steps, K_1 and K_2, seem to be favorable, especially the second, with an elementary equilibrium constant considerably higher than one. The conversion step, K_3, seems to be favorable in general, while the product release steps, K_{4-7} seem to be unfavorable with a few exceptions. Using a cutoff distance of 40 we distinguish five clusters in Fig. 2.6E. In cluster two, the release of the first F6P molecule, K_4, is more favorable than in the other clusters. The same is true for the release of the first PI molecule, K_6, in cluster three. Also the conversion step, K_3, seems to be more favorable in clusters three and four, and the release of the second PI molecule, K_7, more favorable in cluster one.

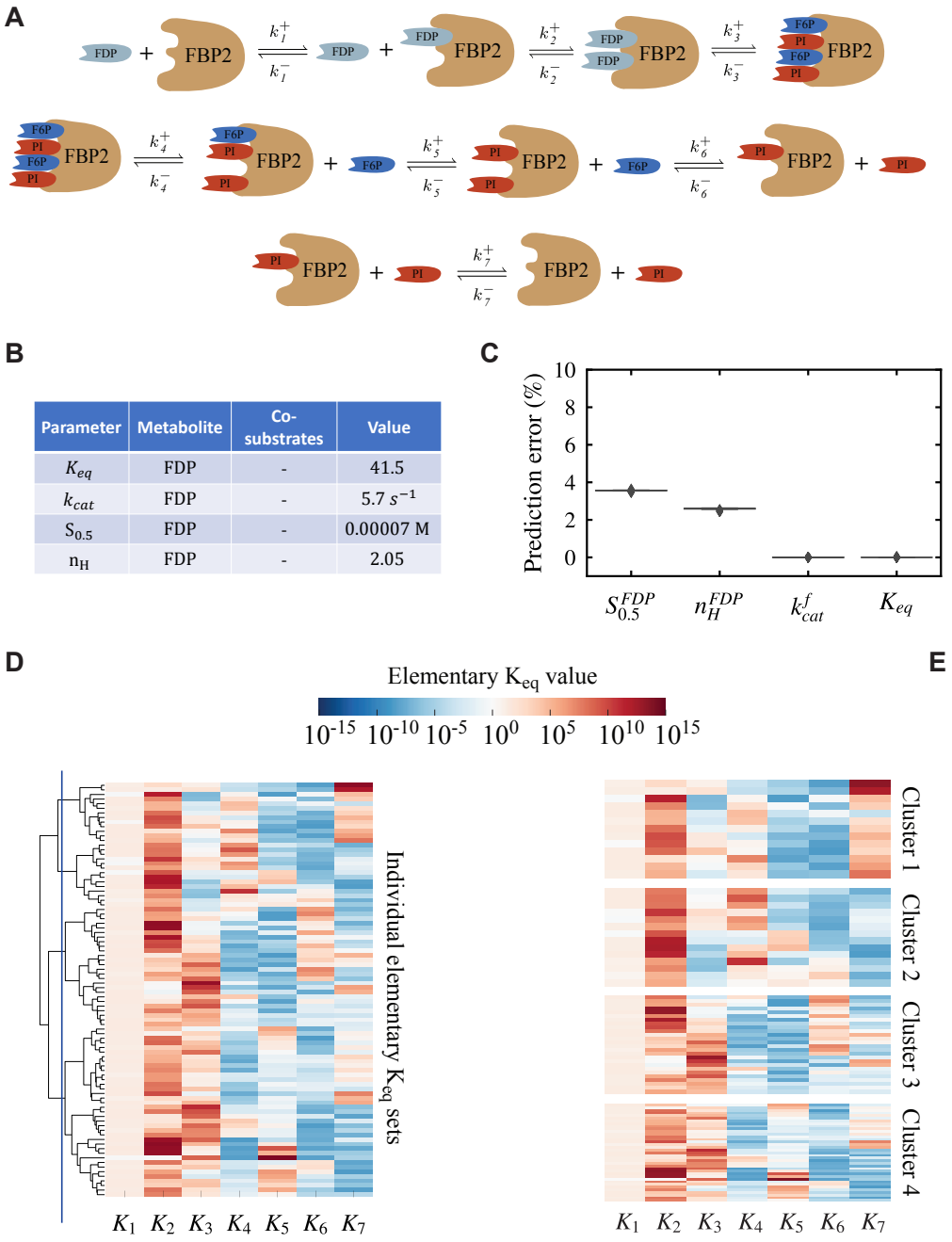


Figure 2.6: (A) Enzyme mechanism for FBP2. (B) Data used to fit the enzyme FBP2. (C) Error associated to the prediction of the kinetic parameters used to fit the elementary rate constants. (D) Clustering of all elementary equilibrium constant sets with ssld< 0.1. (E) Clusters obtained with a cutoff distance of 40.

The Haldane relation used to fit the K_{eq} is the following:

$$K_{eq} = \frac{\overrightarrow{k}_1 \overrightarrow{k}_2 \overrightarrow{k}_3 \overrightarrow{k}_4 \overrightarrow{k}_5 \overrightarrow{k}_6 \overrightarrow{k}_7}{\overleftarrow{k}_1 \overleftarrow{k}_2 \overleftarrow{k}_3 \overleftarrow{k}_4 \overleftarrow{k}_5 \overleftarrow{k}_6 \overleftarrow{k}_7}$$

The equations used to fit the effective forward k_{cat} and the $S_{0.5}^{FDP}$ can be found in section 2.4.2.2.

The Hill coefficient is fitted by simulating data for the $S_{0.5}^{FDP}$ using the Hill equation. For FBP2 the simulated curve to be fitted is shown in Fig. 2.7.

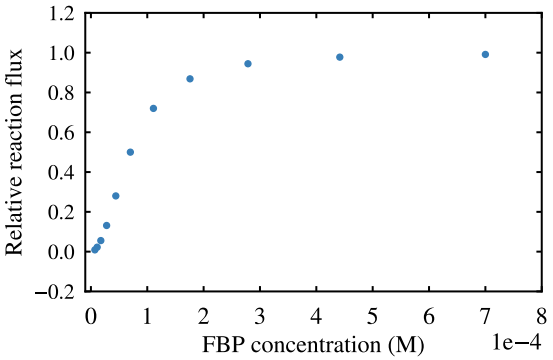


Figure 2.7: Simulated data to fit the $S_{0.5}$ for FDP and respective n_H.

The predicted Hill coefficient is then found by fitting the resulting predicted curve for $S_{0.5}^{FDP}$.

Regarding the results reproducibility, we repeat the enzyme 100 fits three times, and bin the resulting rate constant sets with an ssld < 0.1. However, there are no patterns common to all the three sets of 100 fits, as in FBA1. This again seems to indicate that with 14 rate constants it requires considerably more fits to find all possible combinations that fit the kinetic data. Again, we assembled enzyme-level ODE models from the rate constant sets with ssld< 0.1, for each of the 3 sets of fits, leading to three model ensembles. The omics data used to assemble the ODE models can be found in section 2.4.9. We simulated the resulting model ensembles as a closed system and, as can be seen in Fig. 2.8, the metabolite time-courses are fairly similar for different model ensembles, where each ensemble is plotted in a different column.

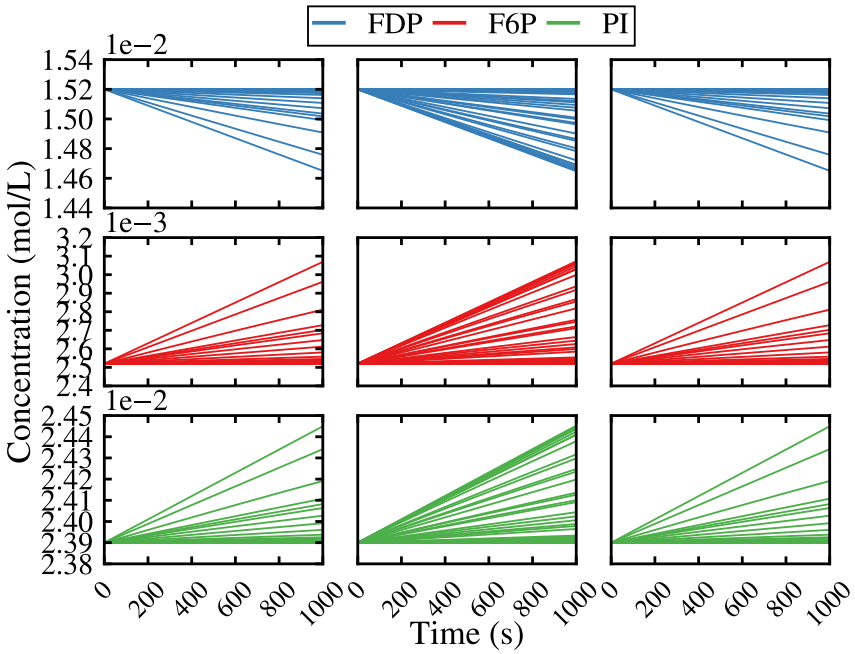


Figure 2.8: Metabolite time-courses for the three model ensembles built through MASSef for FBP2. Each column represents a different model ensemble, and each row a different metabolite, from top to bottom: FDP, F6P, and PI. Each line corresponds to one model. There are a total of 90 models per ensemble.

Finally, we compare the metabolite time-courses obtained through our approach, with those that would be obtained by simulating the enzymatic reaction with a simple Hill rate law. Yet, this comparison is not straightforward. Given the kinetic data available in the literature, we would only be able to use the irreversible Hill rate law. But this wouldn't be a fair comparison, since our model generated through MASSef is modeling a reversible reaction. Thus, we used the reversible Hill rate law for one substrate/two products reactions developed in [17] to compare the model simulations:

$$v = \frac{[FBP2]\, k_{cat} \frac{[FDP]}{s_{0.5}^{FDP}} \left(\frac{[FDP]}{s_{0.5}^{FDP}} + \frac{[F6P][PI]}{s_{0.5}^{F6P} s_{0.5}^{PI}} \right)^{n_H - 1} \left(1 - \frac{[F6P][PI]}{K_{eq}^{FBP2}[FDP]} \right)}{1 - 2\left(\frac{[FDP]}{s_{0.5}^{FDP}} \right)^{n_H} + \left(\frac{[FDP]}{s_{0.5}^{FDP}} + \frac{[F6P]}{s_{0.5}^{F6P}} \right)^{n_H} + \left(\frac{[FDP]}{s_{0.5}^{FDP}} + \frac{[PI]}{s_{0.5}^{PI}} \right)^{n_H} + \left(\frac{[FDP]}{s_{0.5}^{FDP}} + \frac{[F6P][PI]}{s_{0.5}^{F6P} s_{0.5}^{PI}} \right)^{n_H}}$$

To define the values of the effective K_m for F6P and PI, we used the respective equations defined in MASSef and integrated the respective rate constant values. We did the same for the parameters found in the literature, so that the uncertainty with which they are predicted is also included in the Hill models. The results are shown

in Fig. 2.9, and the metabolite time-courses are similar to the ones in Fig. 2.8, even if the Hill models seem to lead to more diverse metabolite time-courses. Note that it would not be possible to simulate the enzyme-level ODE models with a reversible Hill rate law without the K_m values inferred through MASSef. In section 2.4.7, we show the metabolite time-courses obtained from simulating the irreversible Hill models. These were built with the parameters obtained from the corresponding models built through MASSef. Moreover, note that we could also model FBP2 as an irreversible reaction with MASSef if we had chosen to do so.

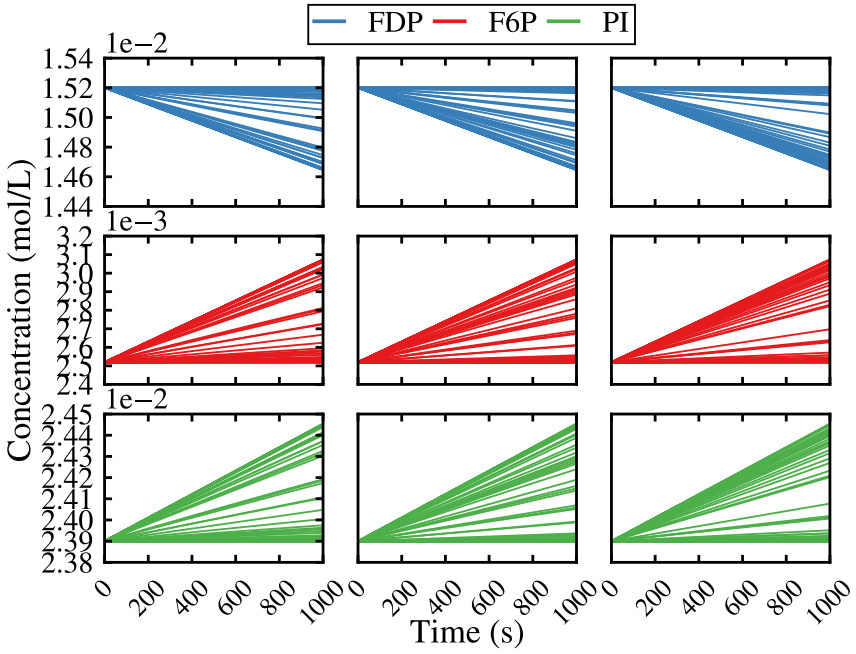


Figure 2.9: Metabolite time-courses obtained from simulating the reversible Hill models for FBP2. Each column represents a model ensemble, build from the parameters inferred through MASSef. Each row represents a different metabolite. Each model ensemble contains 90 models.

Case study 4: Handling inhibition in Fructose-bisphosphate Aldolase class 2 (FBA2)

Finally, we use MASSef to handle both competitive and mixed inhibition in FBA2. FBA2 shows product inhibition by both DHAP and G3P with regards to FDP. In particular, DHAP is a competitive inhibitor with regards to FDP, while G3P is both a competitive and uncompetitive inhibitor with regards to FDP. For the enzyme mechanism and the inhibition reactions see Fig. 2.10A. In Fig. 2.10B we show the

kinetic data to be fitted, where: K_{ic} stands for the competitive inhibition constant of DHAP with regards to FDP; K_{incc} stands for the competitive inhibition constant of G3P with regards to FDP, as part of the mixed inhibition; K_{incu} stands for the uncompetitive inhibition constant of G3P with regards to FDP, as part of the mixed inhibition.

For FBA2 we fitted the enzyme 200 times to get 51 rate constant sets with a ssld lower than 0.0035. In this case it means the largest error between the fitted data points and respective predictions is below 27%. Although most rate constant sets lead to predictions an error below 10%, as shown in Fig. 2.10C.

To fit the K_{eq} we used the following Haldane relation:

$$K_{eq} = \frac{\overrightarrow{k}_1 \overrightarrow{k}_2 \overrightarrow{k}_3 \overrightarrow{k}_4}{\overleftarrow{k}_1 \overleftarrow{k}_2 \overleftarrow{k}_3 \overleftarrow{k}_4}$$

The equations used to fit the k_{cat}, K_m, and the inhibition constants can be found in section 2.4.2.3.

Since this enzyme model includes 14 rate constants, in Fig. 2.10D we show only the clustering results for the elementary equilibrium constants. Overall product release, K_3 and K_4, seems to be unfavorable, while substrate binding, K_1, and the conversion step, K_2, seem to be favorable. Using a maximum distance of 48 for the hierarchical clustering, we distinguish four clusters. In clusters three and four the competitive binding of G3P, K_{incc}^{G3P}, seems more favorable than in cluster one and two. In cluster one both the competitive binding of G3P, K_{incc}^{G3P}, and the binding of FDP to the enzyme-G3P complex, K_{nc}^{G3P}, seem less favorable than in other clusters, while in cluster two the uncompetitive binding of G3P, K_{inccu}^{G3P}, seems less favorable than in the other clusters.

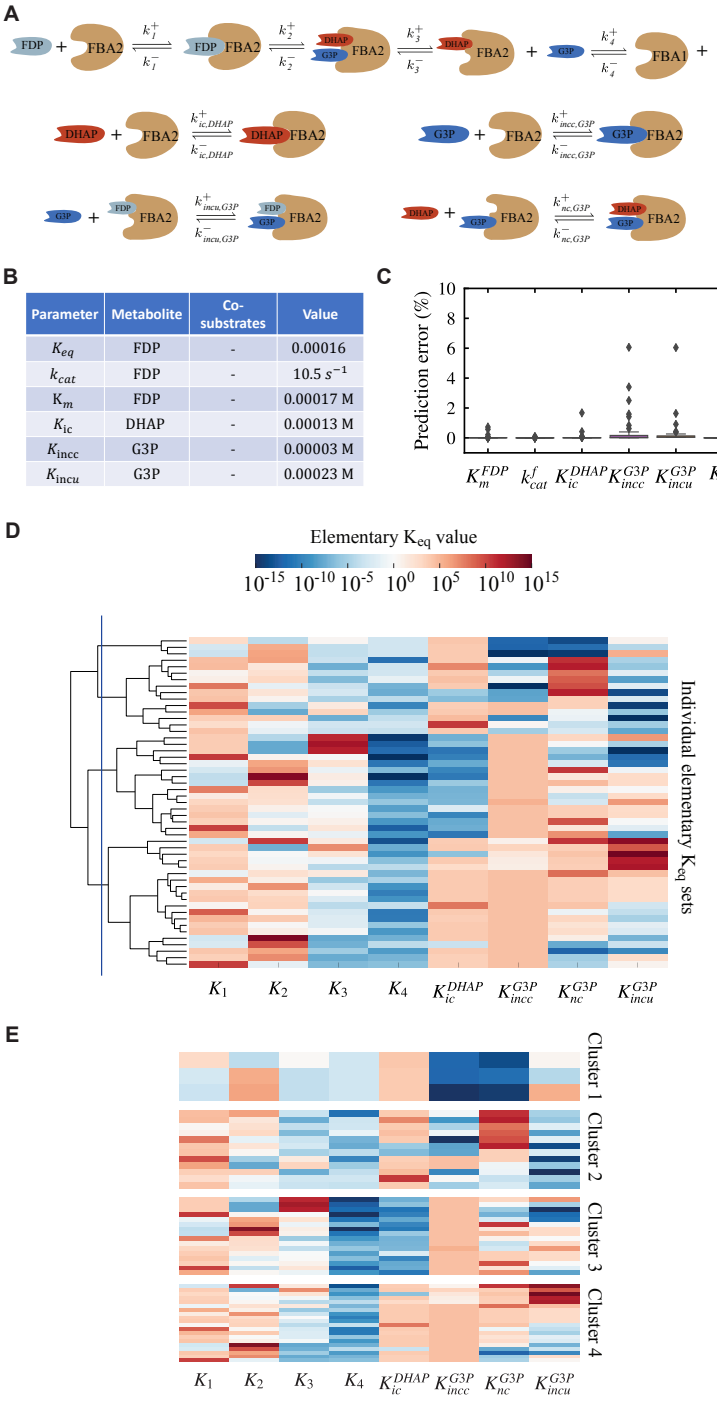


Figure 2.10: (A) Enzyme mechanism for FBA2 in the first row, and the inhibition reactions in the last two rows. (B) Data used to fit the enzyme FBA2. (C) Error associated to the prediction of the kinetic parameters used to fit the elementary rate constants. (D) Clustering of all rate elementary equilibrium sets with ssld$<$ 0.0035. (E) Clusters obtained with a cutoff distance of 40. K_{ic} stands for the competitive inhibition constant of DHAP with regards to FDP; K_{incc} stands for the competitive inhibition constant of G3P with regards to FDP; K_{incu} stands for the uncompetitive inhibition constant of G3P with regards to FDP.

To assess the results reproducibility, we do three sets of 200 fits and bin the resulting rate constant sets. As for FBA1 and FBP2, there are no overlaps between the three sets of fits. Again indicating that for enzymes with a high number of elementary rate constants a larger number of rate constant sets needs to be fitted in order to sample the space of equivalent optima.

2.3 Discussion and conclusions

In this work, we presented eMASS, a workflow for the parameterization of detailed kinetic models of enzyme reactions using enzyme kinetic data, and the respective software tool, MASSef. This software tool has a number of powerful features. First, the workflow allows flexible user-defined reaction mechanisms and can handle the majority of common reaction schemes, including different binding orders, reaction mechanisms such as ping-pong mechanisms, inhibition schemes, protonation reactions, and, in principle, allostery. Second, the workflow enables the fitting of standard kinetic data types, including thermodynamic and initial rate data, and can in principle correct these data to *in vivo*-like conditions. Third, the optimization procedure can handle the inherently highly nonlinear least-squares optimization and sample rate constant sets to deal with under-determined systems. Finally, the workflow is able to model the effect of enzyme inhibitors and activators even when inhibition/activation constants are not available. For instance, if a k_{cat} is measured in the presence and absence of an activator, we can fit both values by defining two catalytic pathways: one where activator is bound to the enzyme for all elementary steps, and one where it is absent. Also, one advantage of this framework is that one does not need to find the appropriate rate law to model each process, instead one needs to specify the appropriate mechanisms. Besides, it can highlight conflicts between different kinetic data.

The modeling procedure here depends on user-specified reaction mechanisms. The reaction schemes that we have tested thus far are those based on standards set by Cleland and others enzymologists. These reaction mechanisms have certain assumptions but have seen practical success in representing enzyme behavior in initial rate and progress curve experiments. There are additional levels of detail that potentially could be represented that we have not yet tested. These include additional breakdown of catalysis into individual interactions between the substrate and catalytic residues as well as breakdown of common deterministic and well-mixed assumptions due to

restricted geometries, low copy numbers, channeling, stochastic behavior, proton tunneling, and other more complex kinetic phenomena. These more complex situations have been handled by others in various ways but not yet integrated into our workflows.

The current workflows have been developed to handle reaction K_{eq} values as well as enzyme initial rate data such as k_{cat}, K_m, and K_i, due to the dominance of these data types in the enzyme kinetic literature. However, in principle this workflow could be extended to additional data types such as progress curve data and stopflow data. The requisite for utilizing these data types is the construction of model equations to be used in the least-squares optimization. For example, to compare to progress rate data, the mass balance equations for the enzyme would be integrated over time and compared to the experimental time course. This is feasible in principle but would have an unknown computational complexity. Additionally, in the current procedure we use simulated experimental curves that are effectively back-calculated for the parameters in the case of K_m and K_i. This procedure was implemented because the original data curves were often not available or were inconvenient to extract from the literature. However, in principle the original data plots could be fit directly and should have mostly equivalent results to the current workflow.

The current workflow attempts to correct for difference between the experimental conditions and *in vivo* environment by correcting for temperature and pH. These correction methods are taken largely from approaches used by enzymologists to model these same effects. We could also correct concentrations to the more accurate chemical activities using a Debye-Huckel model for the activity coefficient given the ionic strength under experimental conditions [34]. However, these methods often require a large amount of data on particular enzymes across various conditions and thus are difficult to apply to the majority of enzymes. Additionally, there are also often a number of unknown effects, such as buffer interactions with enzyme or macromolecular crowding effects *in vivo*, that we currently cannot account for. Thus, there appears to be much work required to continue to improve our ability to utilize *in vitro* enzyme kinetic assay data to model *in vivo* systems. Both sensitivity analysis and adequate benchmarking will be critical to determine which of these corrections are important and can be performed with sufficient accuracy.

Mass action enzyme systems have many more rate constants than kinetic data typically, and thus are highly underdetermined. One of the most powerful aspects of this workflow is the ability to fit rate constants to measured kinetic data without requiring that these rate constants are fully specified, due to the inherent randomized parameter sampling built into the optimization procedure. This is enabled by the

pseudo-random start points for the LMA optimization that are provided by the initial non-derivative based PSO optimization. One potential issue is to determine how completely the parameter space is being sampled. We attempt to address this by both clustering the rate constants through successive optimizations and binning each rate constant set, and considering the parameter space to be fully specified when new clusters or new rate constant set patterns are no longer found. However, this procedure cannot discount inherent bias in the optimization procedure that would lead parameters with particular biases to be oversampled.

Kinetic modeling of metabolic networks has seen a resurgence in recent years, driven largely by parameterization strategies involving sampling and system-level parameter fitting. The work here is intended to increase the accessibility of a 'bottom-up' parameterization approach that makes use of the plethora of historical enzyme kinetic data available in the literature. As these models are parameterized enzyme-by-enzyme, this approach is expected to scale well to the network scale. Additionally, there have been efforts recently to fill gaps in critical parameters such as turnover rates at the genome-scale that should further enable these efforts. As parameterization becomes increasingly computationally feasible and biochemically accurate, practical kinetic models will likely soon become commonly accessible and powerful tools available to the systems biology community.

2.4 Supplementary information

2.4.1 Experimental data used in the case studies

2.4.1.1 TPI

Parameter	Weight	Metabolite	Co-substrate	Value	T (°C)	pH	Buffer	Salts	Reference (PMID)
K_{eq}	1	G3P	-	9.35		7.5	-	-	eQuilibrator
k_{cat}	1	G3P	-	27025.3 s^{-1}	37	7.6	teoahcl, 0.1 M	-	9442062
K_m	1	G3P	-	0.00103 M	25	7.6	teoahcl, 0.1 M	-	9442062

2.4.1.2 FBA1

Parameter	Weight	Metabolite	Co-substrate	Value	T (°C)	pH	Buffer	Salts	Reference (PMID)
K_{eq}	1	FDP	-	0.00016	-	7.5	-	-	eQuilibrator
k_{cat}	2	FDP	-	0.41 s^{-1}	37	8	trishcl, 0.05 M	-	9531482
k_{cat}	1	FDP	citrate, 0.011 M	6 s^{-1}	37	8	trishcl, 0.05 M	-	9531482
K_m	10	FDP	-	0.00002 M	30	8	trishcl, 0.05 M	-	9531482
K_m	10	FDP	citrate, 0.011 M	0.00007 M	30	8	trishcl, 0.05 M	-	9531482

2.4.1.3 FBP2

Parameter	Weight	Metabolite	Co-substrate	Value	T (°C)	pH	Buffer	Salts	Reference (PMID)
K_{eq}	1	FDP	-	41.5	-	7.5	-	-	eQuilibrator
k_{cat}	1	FDP	-	5.7 s^{-1}	37	9	chesk, 0.5 M	Mn^{2+}, 0.02M, Cl^{-}, 0.04 M	19073594
$S_{0.5}$	1	FDP	-	0.00007 M	37	9	chesk, 0.5 M	Mn^{2+}, 0.02M, Cl^{-}, 0.04 M	19073594
n_H	1	FDP		20.5	37	9	chesk, 0.5 M	Mn^{2+}, 0.02M, Cl^{-}, 0.04 M	19073594

2.4.1.4 FBA2

Parameter	Weight	Metabolite	Co-substrate	Value	T (°C)	pH	Buffer	Salts	Reference (PMID)
K_{eq}	1	FDP	-	0.00016		7.5		-	eQuilibrator
k_{cat}	1	FDP	-		37	8	trishcl, 0.05 M	-	10712619
K_m	1	FDP		0.00017 M	30	8	trishcl, 0.05 M	-	10712619
K_{ic}	1	DHAP		0.00013 M	30	8	trishcl, 0.05 M	-	10712619
K_{incc}	1	G3P	-	0.00003 M	30	8	trishcl, 0.05 M	-	10712619
K_{incu}	1	G3P		0.00023 M	30	8	trishcl, 0.05 M	-	10712619

2.4.2 Equations used to fit data in the case studies

2.4.2.1 FBA1

Given the elementary reactions:

$$\left\{\overset{\text{FBA11}}{\text{FBA1}^{c} + \text{cit}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{cit}^{c}},\ \overset{\text{FBA12}}{\text{FBA1}^{c} + \text{fdp}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{fdp}^{c}},\right.$$

$$\overset{\text{FBA13}}{\text{FBA1}^{c}\ \&\ \text{cit}^{c} + \text{fdp}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{fdp}^{c}},\ \overset{\text{FBA14}}{\text{FBA1}^{c}\ \&\ \text{dhap}^{c} \rightleftharpoons \text{FBA1}^{c} + \text{dhap}^{c}},$$

$$\overset{\text{FBA15}}{\text{FBA1}^{c}\ \&\ \text{fdp}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{dhap}^{c}\ \&\ \text{g3p}^{c}},\ \overset{\text{FBA16}}{\text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{dhap}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{cit}^{c} + \text{dhap}^{c}},$$

$$\overset{\text{FBA17}}{\text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{fdp}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{dhap}^{c}\ \&\ \text{g3p}^{c}},$$

$$\overset{\text{FBA18}}{\text{FBA1}^{c}\ \&\ \text{dhap}^{c}\ \&\ \text{g3p}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{dhap}^{c} + \text{g3p}^{c}},$$

$$\left.\overset{\text{FBA19}}{\text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{dhap}^{c}\ \&\ \text{g3p}^{c} \rightleftharpoons \text{FBA1}^{c}\ \&\ \text{cit}^{c}\ \&\ \text{dhap}^{c} + \text{g3p}^{c}}\right\}$$

the equation used to fit the forward k_{cat} is:

$$\left(\mathrm{fdp}^{c}\ \underset{\text{Global}}{\mathrm{FBA1\_total}}\ k^{\rightarrow}_{FBA14}\ k^{\rightarrow}_{FBA16}\right.$$
$$\left(\mathrm{cit}^{c}\ k^{\rightarrow}_{FBA11}\ k^{\rightarrow}_{FBA13}\ k^{\rightarrow}_{FBA17}\ \left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)\ k^{\rightarrow}_{FBA19} + \right.$$
$$\left.\left.k^{\leftarrow}_{FBA11}\ k^{\rightarrow}_{FBA12}\ k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18}\ \left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right)\right)\right) /$$
$$\left(k^{\leftarrow}_{FBA11}\ \left(k^{\rightarrow}_{FBA14}\ k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right) + \right.\right.$$
$$\left.\mathrm{fdp}^{c}\ k^{\rightarrow}_{FBA12}\ \left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)\right)$$
$$\left(k^{\rightarrow}_{FBA16}\ k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right) +$$
$$\mathrm{cit}^{c}\ k^{\rightarrow}_{FBA11}\ \left(k^{\rightarrow}_{FBA14}\ k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)$$
$$\left(k^{\rightarrow}_{FBA16}\ k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right) + \right.$$
$$\left.\left.\mathrm{fdp}^{c}\ k^{\rightarrow}_{FBA13}\ \left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right)\right)\right)$$

and the equation used to fit the K_m for FDP is:

$$\left(\mathrm{fdp}^{c}\ \left(k^{\leftarrow}_{FBA11}\ k^{\rightarrow}_{FBA16}\ \left(k^{\rightarrow}_{FBA14}\ k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right) + \right.\right.\right.$$
$$\left.\left.k^{\rightarrow}_{FBA12}\ \left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)\right)$$
$$\left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right) + \mathrm{cit}^{c}\ k^{\rightarrow}_{FBA11}\ k^{\rightarrow}_{FBA14}$$
$$\left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)$$
$$\left(k^{\rightarrow}_{FBA16}\ k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right) + \right.$$
$$\left.\left.\left.k^{\rightarrow}_{FBA13}\ \left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right)\right)\right)\right) /$$
$$\left(k^{\leftarrow}_{FBA11}\ k^{\rightarrow}_{FBA16}\ \left(k^{\rightarrow}_{FBA14}\ k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right) + \right.\right.$$
$$\left.\mathrm{fdp}^{c}\ k^{\rightarrow}_{FBA12}\ \left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\rightarrow}_{FBA14}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)\right)$$
$$\left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right) + \mathrm{cit}^{c}\ k^{\rightarrow}_{FBA11}\ k^{\rightarrow}_{FBA14}$$
$$\left(k^{\rightarrow}_{FBA15}\ k^{\rightarrow}_{FBA18} + k^{\leftarrow}_{FBA12}\ \left(k^{\leftarrow}_{FBA15} + k^{\rightarrow}_{FBA18}\right)\right)$$
$$\left(k^{\rightarrow}_{FBA16}\ k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\leftarrow}_{FBA13}\ k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right) + \right.$$
$$\left.\left.\mathrm{fdp}^{c}\ k^{\rightarrow}_{FBA13}\ \left(k^{\rightarrow}_{FBA17}\ k^{\rightarrow}_{FBA19} + k^{\rightarrow}_{FBA16}\ \left(k^{\leftarrow}_{FBA17} + k^{\rightarrow}_{FBA17} + k^{\rightarrow}_{FBA19}\right)\right)\right)\right)$$

2.4.2.2 FBP2

Considering the elementary reactions for FBP2:

$$\left\{\mathrm{FBP2^c} + \mathrm{fdp^c} \overset{\mathrm{FBP21}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ fdp^c},\ \mathrm{FBP2^c\ \&\ pi^c} \overset{\mathrm{FBP22}}{\rightleftharpoons} \mathrm{FBP2^c} + \mathrm{pi^c},\right.$$

$$\mathrm{FBP2^c\ \&\ fdp^c} + \mathrm{fdp^c} \overset{\mathrm{FBP23}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ fdp^c\ \&\ fdp^c},\ \mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c} \overset{\mathrm{FBP24}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ pi^c} + \mathrm{pi^c},$$

$$\mathrm{FBP2^c\ \&\ fdp^c\ \&\ fdp^c} \overset{\mathrm{FBP25}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c\ \&\ f6p^c\ \&\ f6p^c},$$

$$\mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c\ \&\ f6p^c} \overset{\mathrm{FBP26}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c} + \mathrm{f6p^c},$$

$$\left.\mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c\ \&\ f6p^c\ \&\ f6p^c} \overset{\mathrm{FBP27}}{\rightleftharpoons} \mathrm{FBP2^c\ \&\ pi^c\ \&\ pi^c\ \&\ f6p^c} + \mathrm{f6p^c}\right\}$$

we use the following equation to fit the forward k_{cat}

$$\left(2\ (\mathrm{fdp^c})^2\ \underset{\mathrm{Global}}{\mathrm{FBP2\_total}}\ \overrightarrow{k_{\mathrm{FBP21}}}\ \overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}}\right) \Big/$$
$$\left(\overleftarrow{k_{\mathrm{FBP21}}}\ \overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overleftarrow{k_{\mathrm{FBP23}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right) +\right.$$
$$\mathrm{fdp^c}\ \left(\overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.$$
$$\overrightarrow{k_{\mathrm{FBP21}}}\ \left(\mathrm{fdp^c}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP22}}}\ \left(\overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.\right.$$
$$\overleftarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right) + \mathrm{fdp^c}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.$$
$$\left.\left.\left.\left.\left.\left.\overrightarrow{k_{\mathrm{FBP24}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP25}}}\ \left(\overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right)\right)\right)\right)\right)\right)$$

and the following equation to fit the $S_{0.5}$ for FDP:

$$\left((\mathrm{fdp^c})^2\ \left(\overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.\right.$$
$$\overleftarrow{k_{\mathrm{FBP21}}}\ \overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overleftarrow{k_{\mathrm{FBP23}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right) +$$
$$\overrightarrow{k_{\mathrm{FBP21}}}\ \left(\overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP22}}}\ \left(\overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.\right.$$
$$\overleftarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right) + \overrightarrow{k_{\mathrm{FBP23}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.$$
$$\left.\left.\left.\left.\left.\left.\overrightarrow{k_{\mathrm{FBP24}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP25}}}\ \left(\overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right)\right)\right)\right)\right)\right) \Big/$$
$$\left(\overleftarrow{k_{\mathrm{FBP21}}}\ \overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overleftarrow{k_{\mathrm{FBP23}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right) +\right.$$
$$\mathrm{fdp^c}\ \left(\overrightarrow{k_{\mathrm{FBP22}}}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.$$
$$\overrightarrow{k_{\mathrm{FBP21}}}\ \left(\mathrm{fdp^c}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP22}}}\ \left(\overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.\right.$$
$$\overleftarrow{k_{\mathrm{FBP23}}}\ \overrightarrow{k_{\mathrm{FBP24}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right) + \mathrm{fdp^c}\ \overrightarrow{k_{\mathrm{FBP23}}}\ \left(\overrightarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} +\right.$$
$$\left.\left.\left.\left.\left.\left.\overrightarrow{k_{\mathrm{FBP24}}}\ \left(\overleftarrow{k_{\mathrm{FBP25}}}\ \overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP26}}}\ \overrightarrow{k_{\mathrm{FBP27}}} + \overrightarrow{k_{\mathrm{FBP25}}}\ \left(\overrightarrow{k_{\mathrm{FBP26}}} + \overrightarrow{k_{\mathrm{FBP27}}}\right)\right)\right)\right)\right)\right)\right)$$

2.4.2.3 FBA2

FBA2 has the following elementary reactions:

$$\left\{\mathrm{FBA2^c} + \mathrm{fdp^c} \overset{\mathrm{FBA21}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ fdp^c},\ \mathrm{FBA2^c\ \&\ dhap^c} \overset{\mathrm{FBA22}}{\rightleftharpoons} \mathrm{FBA2^c} + \mathrm{dhap^c},\right.$$

$$\mathrm{FBA2^c\ \&\ fdp^c} \overset{\mathrm{FBA23}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ dhap^c\ \&\ g3p^c},\ \mathrm{FBA2^c\ \&\ dhap^c\ \&\ g3p^c} \overset{\mathrm{FBA24}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ dhap^c} + \mathrm{g3p^c},$$

$$\mathrm{FBA2^c} + \mathrm{dhap^c} \overset{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ dhap^c},\ \mathrm{FBA2^c} + \mathrm{g3p^c} \overset{\mathrm{FBA2\_Kincc\_g3p\_1\_fdp}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ g3p^c},$$

$$\left.\mathrm{FBA2^c\ \&\ g3p^c} + \mathrm{fdp^c} \overset{\mathrm{FBA2\_NC\_g3p}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ fdp^c\ \&\ g3p^c},\ \mathrm{FBA2^c\ \&\ fdp^c} + \mathrm{g3p^c} \overset{\mathrm{FBA2\_Kincu\_g3p\_1\_fdp}}{\rightleftharpoons} \mathrm{FBA2^c\ \&\ fdp^c\ \&\ g3p^c}\right\}$$

Which lead to the following equation to fit the forward k_{cat}:

$$\left(\mathrm{fdp^c}\ \underset{\mathrm{Global}}{\mathrm{FBA2\_total}}\ k^{\rightarrow}_{\mathrm{FBA21}}\ k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\rightarrow}_{\mathrm{FBA24}}\ \left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right) \Big/$$
$$\left(\mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\rightarrow}_{\mathrm{FBA24}} + \right.$$
$$k^{\rightarrow}_{\mathrm{FBA22}}\ \left(k^{\rightarrow}_{\mathrm{FBA23}}\ \left(\mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}} + k^{\rightarrow}_{\mathrm{FBA24}}\right) + \mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\right) +$$
$$\mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ k^{\leftarrow}_{\mathrm{FBA23}}\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} + \mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} +$$
$$\mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ k^{\rightarrow}_{\mathrm{FBA24}}\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} + k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\rightarrow}_{\mathrm{FBA24}}\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} +$$
$$\left.k^{\leftarrow}_{\mathrm{FBA21}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\ \left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)$$

the equation to fit the K_m for FDP:

$$\left\{\left(\mathrm{fdp^c}\ \left(k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\rightarrow}_{\mathrm{FBA24}}\ \left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right) + \right.\right.$$
$$k^{\leftarrow}_{\mathrm{FBA21}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\ \left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right) +$$
$$k^{\rightarrow}_{\mathrm{FBA21}}\ \left(k^{\rightarrow}_{\mathrm{FBA22}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right) + \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} +$$
$$\left.\left.\left.k^{\rightarrow}_{\mathrm{FBA23}}\ \left(k^{\rightarrow}_{\mathrm{FBA24}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)\right)\right) \Big/$$
$$\left(k^{\rightarrow}_{\mathrm{FBA23}}\ k^{\rightarrow}_{\mathrm{FBA24}}\ \left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right) + k^{\leftarrow}_{\mathrm{FBA21}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\right.$$
$$\left(k^{\rightarrow}_{\mathrm{FBA22}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right) + \mathrm{fdp^c}\ k^{\rightarrow}_{\mathrm{FBA21}}\ \left(k^{\rightarrow}_{\mathrm{FBA22}}\ \left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right) +$$
$$\left.\left.\left.\left(k^{\leftarrow}_{\mathrm{FBA23}} + k^{\rightarrow}_{\mathrm{FBA24}}\right)\ k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}} + k^{\rightarrow}_{\mathrm{FBA23}}\ \left(k^{\rightarrow}_{\mathrm{FBA24}} + k^{\leftarrow}_{\mathrm{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)\right)\right\}$$

the equation to fit the competitive inhibition by DHAP, which is generated by dividing v_{ss} for FBA2 by the respective $V_{max,eff}$ (v_{ss} when the substrate concentration is 1 M) and setting only [G3P] to null:

$$\left(\mathrm{fdp}^{c}\left(\overrightarrow{k_{FBA23}}\,\overrightarrow{k_{FBA24}}\left(\overrightarrow{k_{FBA22}}+\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}\right)+\overleftarrow{k_{FBA21}}\left(\overleftarrow{k_{FBA23}}+\overrightarrow{k_{FBA24}}\right)\left(\overrightarrow{k_{FBA22}}+\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}\right)+\overrightarrow{k_{FBA21}}\left(\overrightarrow{k_{FBA22}}\left(\overleftarrow{k_{FBA23}}+\overrightarrow{k_{FBA23}}+\overrightarrow{k_{FBA24}}\right)+\left(\overleftarrow{k_{FBA23}}+\overrightarrow{k_{FBA24}}\right)\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}+\overrightarrow{k_{FBA23}}\left(\overrightarrow{k_{FBA24}}+\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)\right)\right)/\left(\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA22}}\,\overleftarrow{k_{FBA23}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA22}}\,\overrightarrow{k_{FBA23}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA22}}\,\overrightarrow{k_{FBA24}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA23}}\,\overrightarrow{k_{FBA24}}+\overrightarrow{k_{FBA22}}\,\overrightarrow{k_{FBA23}}\,\overrightarrow{k_{FBA24}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overleftarrow{k_{FBA23}}\,\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA23}}\,\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}+\mathrm{fdp}^{c}\,\overrightarrow{k_{FBA21}}\,\overrightarrow{k_{FBA24}}\,\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}+\overrightarrow{k_{FBA23}}\,\overrightarrow{k_{FBA24}}\,\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}+\overleftarrow{k_{FBA21}}\left(\overleftarrow{k_{FBA23}}+\overrightarrow{k_{FBA24}}\right)\left(\overrightarrow{k_{FBA22}}+\overleftarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}\right)+\mathrm{dhap}^{c}\left(\overrightarrow{k_{FBA23}}\,\overrightarrow{k_{FBA24}}+\overleftarrow{k_{FBA21}}\left(\overleftarrow{k_{FBA23}}+\overrightarrow{k_{FBA24}}\right)\right)\left(\overleftarrow{k_{FBA22}}+\overrightarrow{k_{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)\},$$

the equation to fit the mixed inhibition by G3P, which is generated by dividing v_{ss} for FBA2 by the respective $V_{max,eff}$ (v_{ss} when the substrate concentration is 1 M) and setting only [DHAP] to null::

$$\left(\mathrm{fdp}^{c}\left(k_{\overrightarrow{FBA23}}\,k_{\overrightarrow{FBA24}}\left(k_{\overrightarrow{FBA22}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)+k_{\overleftarrow{FBA21}}\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)\left(k_{\overrightarrow{FBA22}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)+k_{\overrightarrow{FBA21}}\right.\right.$$
$$\left.\left(k_{\overrightarrow{FBA22}}\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)+\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}+k_{\overrightarrow{FBA23}}\left(k_{\overrightarrow{FBA24}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)\right)$$
$$\left(\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overrightarrow{FBA2\_NC\_g3p}}+\right.$$
$$\left.k_{\overrightarrow{FBA21}}\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+\mathrm{fdp}^{c}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overrightarrow{FBA2\_NC\_g3p}}\right)\right)\Big/$$
$$\left(k_{\overrightarrow{FBA21}}\left(\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}+\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\right)\right.\right.$$
$$\left(\mathrm{g3p}^{c}\left(\mathrm{g3p}^{c}\,k_{\overleftarrow{FBA23}}\,k_{\overleftarrow{FBA24}}+\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overleftarrow{FBA2\_NC\_g3p}}+\right.$$
$$k_{\overrightarrow{FBA23}}\,k_{\overrightarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+k_{\overrightarrow{FBA22}}\left(\mathrm{g3p}^{c}\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)\right.$$
$$\left.\left.k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overleftarrow{FBA2\_NC\_g3p}}+k_{\overrightarrow{FBA23}}\,k_{\overrightarrow{FBA24}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)\right)\right)+\left(\mathrm{fdp}^{c}\right)^{2}k_{\overrightarrow{FBA21}}$$
$$\left(\left(\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}+k_{\overrightarrow{FBA23}}\left(k_{\overrightarrow{FBA24}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\left(\mathrm{g3p}^{c}\right)^{2}\right.$$
$$k_{\overleftarrow{FBA23}}\,k_{\overleftarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overrightarrow{FBA22}}\left(k_{\overrightarrow{FBA23}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA23}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\right.\right.$$
$$\left.\left.\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)+k_{\overrightarrow{FBA24}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)\right)+$$
$$\mathrm{g3p}^{c}\left(k_{\overrightarrow{FBA23}}\,k_{\overleftarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overrightarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\right.$$
$$\left.\left.k_{\overleftarrow{FBA23}}\left(k_{\overleftarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)\right)\right)k_{\overrightarrow{FBA2\_NC\_g3p}}+$$
$$\mathrm{fdp}^{c}\left(k_{\overrightarrow{FBA21}}\left(\left(\mathrm{g3p}^{c}\right)^{2}k_{\overleftarrow{FBA23}}\,k_{\overleftarrow{FBA24}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+\right.\right.$$
$$\left(\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}+k_{\overrightarrow{FBA23}}\left(k_{\overrightarrow{FBA24}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)\right)k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}$$
$$\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+\mathrm{g3p}^{c}\left(k_{\overrightarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right.$$
$$\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+k_{\overrightarrow{FBA23}}\,k_{\overrightarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\right.$$
$$\left.k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+k_{\overrightarrow{FBA23}}\left(k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}+\right.\right.$$
$$\left.\left.\left.k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+k_{\overrightarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)\right)\right)+$$
$$k_{\overrightarrow{FBA22}}\left(k_{\overrightarrow{FBA23}}\,k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+k_{\overleftarrow{FBA23}}\right.$$
$$\left(\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overleftarrow{FBA2\_NC\_g3p}}+k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\right.\right.$$
$$\left.\left.\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)\right)+k_{\overrightarrow{FBA24}}\left(\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overrightarrow{FBA2\_NC\_g3p}}+\right.$$
$$\left.\left.k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)\right)\right)\right)+$$
$$\left(k_{\overrightarrow{FBA23}}\,k_{\overrightarrow{FBA24}}\left(k_{\overrightarrow{FBA22}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\right.$$
$$\left(\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\right.$$
$$k_{\overrightarrow{FBA22}}\left(\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overrightarrow{FBA23}}\left(k_{\overrightarrow{FBA24}}+k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)\right)+k_{\overrightarrow{FBA23}}$$
$$\left.\left(k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overrightarrow{FBA24}}\left(k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}+k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)\right)\right)+$$
$$\left(\mathrm{g3p}^{c}\right)^{3}k_{\overleftarrow{FBA23}}\,k_{\overleftarrow{FBA24}}\,k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\,k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\left(\mathrm{g3p}^{c}\right)^{2}k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}$$
$$\left(k_{\overrightarrow{FBA23}}\,k_{\overleftarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overrightarrow{FBA24}}\left(k_{\overrightarrow{FBA22}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA23}}\right.$$
$$\left.\left.\left(k_{\overleftarrow{FBA24}}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+\left(k_{\overrightarrow{FBA22}}+k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)k_{\overrightarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\right)\right)k_{\overrightarrow{FBA2\_NC\_g3p}}\right)+$$
$$k_{\overleftarrow{FBA21}}\left(\mathrm{g3p}^{c}\,k_{\overleftarrow{FBA23}}\,k_{\overleftarrow{FBA24}}+k_{\overrightarrow{FBA22}}\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)+\left(k_{\overleftarrow{FBA23}}+k_{\overrightarrow{FBA24}}\right)k_{\overleftarrow{FBA2\_Kic\_dhap\_1\_fdp}}\right)$$
$$\left(k_{\overleftarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}\right)+\mathrm{fdp}^{c}\,k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}\,k_{\overrightarrow{FBA2\_NC\_g3p}}+\right.$$
$$\left.\left.\left.\mathrm{g3p}^{c}\,k_{\overrightarrow{FBA2\_Kincc\_g3p\_1\_fdp}}\left(k_{\overleftarrow{FBA2\_Kincu\_g3p\_1\_fdp}}+k_{\overleftarrow{FBA2\_NC\_g3p}}+\mathrm{fdp}^{c}\,k_{\overrightarrow{FBA2\_NC\_g3p}}\right)\right)\right)\right)\right\}\right\}$$

2.4.3 Rate constants clustermaps for TPI

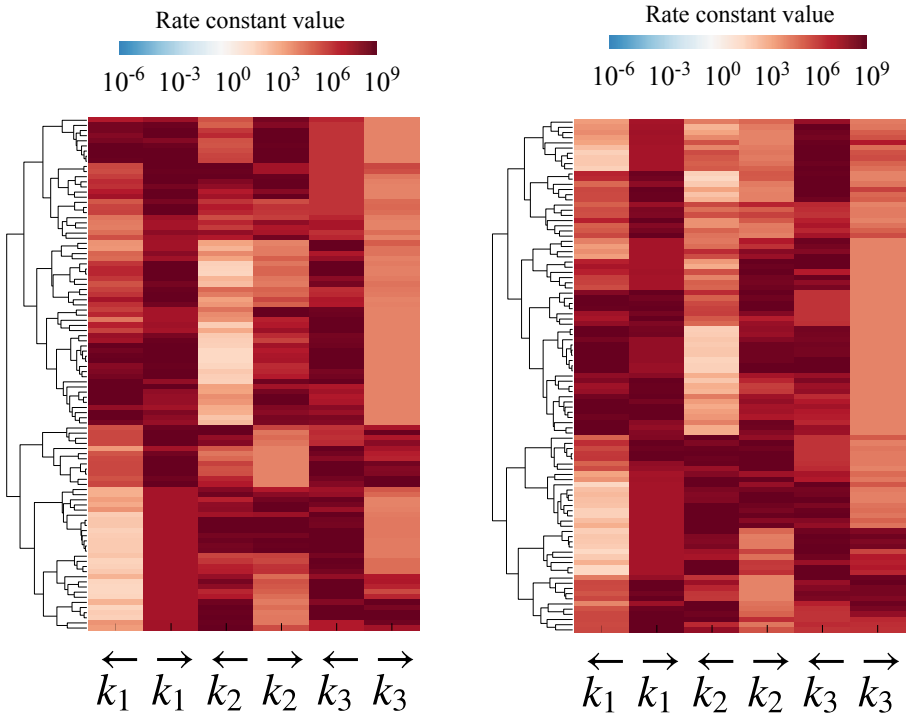
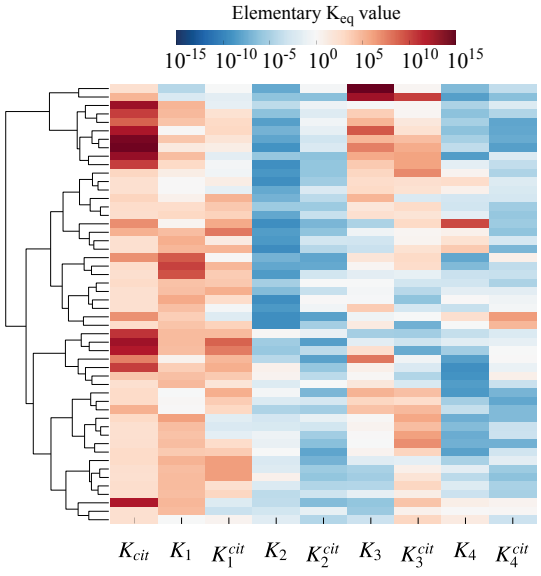
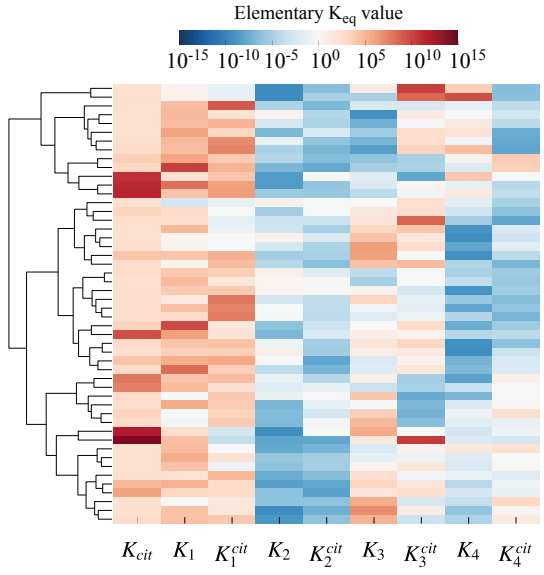


Figure 2.11: Rate constant clustering results for two out of three sets of 100 fits for TPI.

2.4.4 Elementary equilibrium constants clustermaps for FBA1



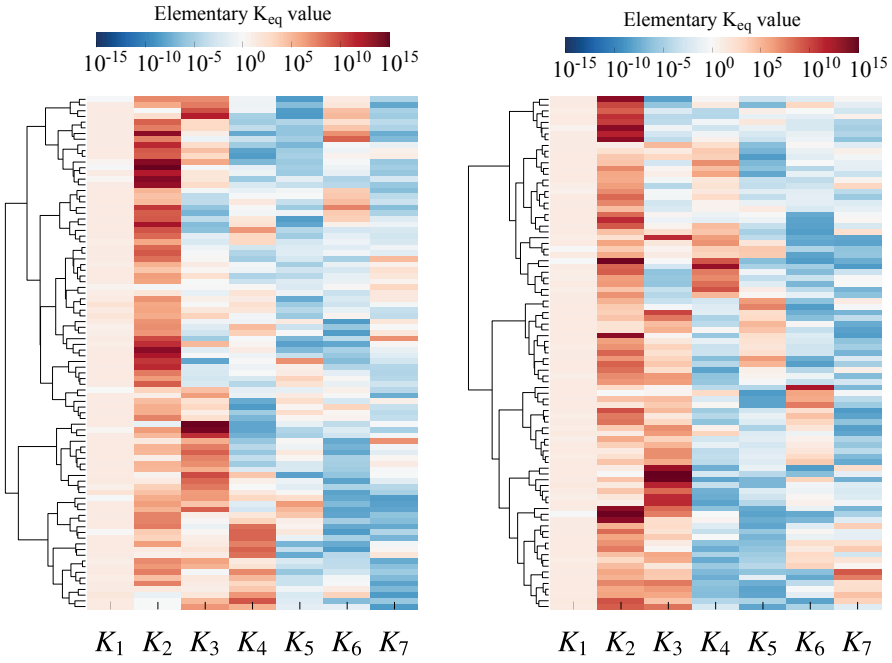
(a)



(b)

Figure 2.12: Elementary equilibrium constant clustering results for two out of three sets of 1000 fits for FBA1. Only elementary equilibrium constant sets with ssld lower than 0.005 are shown.

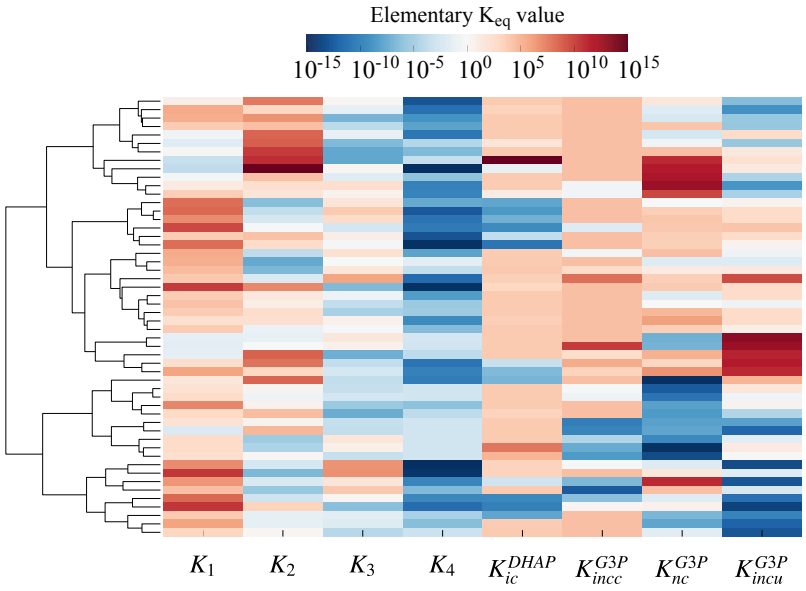
2.4.5 Elementary equilibrium constants clustermaps for FBP2



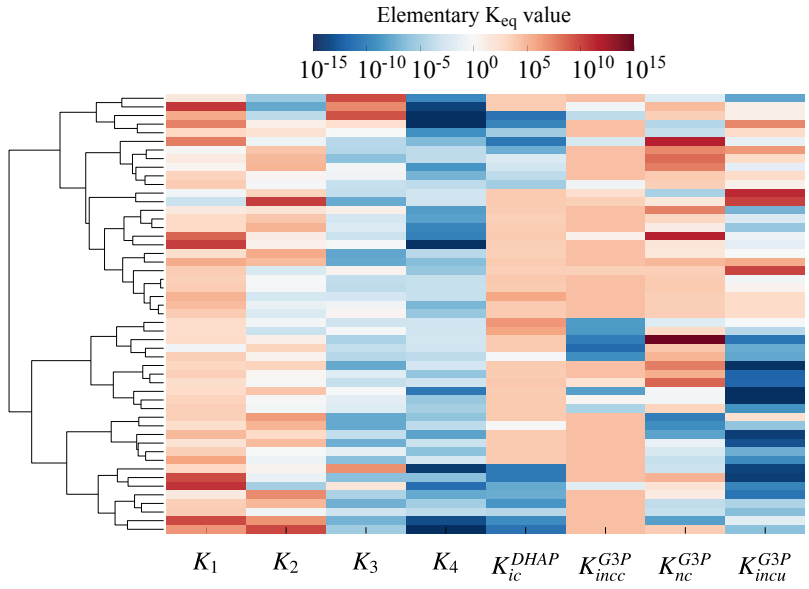
(a) (b)

Figure 2.13: Elementary equilibrium constant clustering results for two out of three sets of 100 fits for FBP2. Only elementary equilibrium constant sets with ssld lower than 0.1 are shown.

2.4.6 Elementary equilibrium constants clustermaps for FBA2



(a)



(b)

Figure 2.14: Elementary equilibrium constant clustering results for two out of three sets of 200 fits for FBA2. Only elementary equilibrium constant sets with ssld lower than 0.0035 are shown.

2.4.7 Irreversible Hill models for FBP2

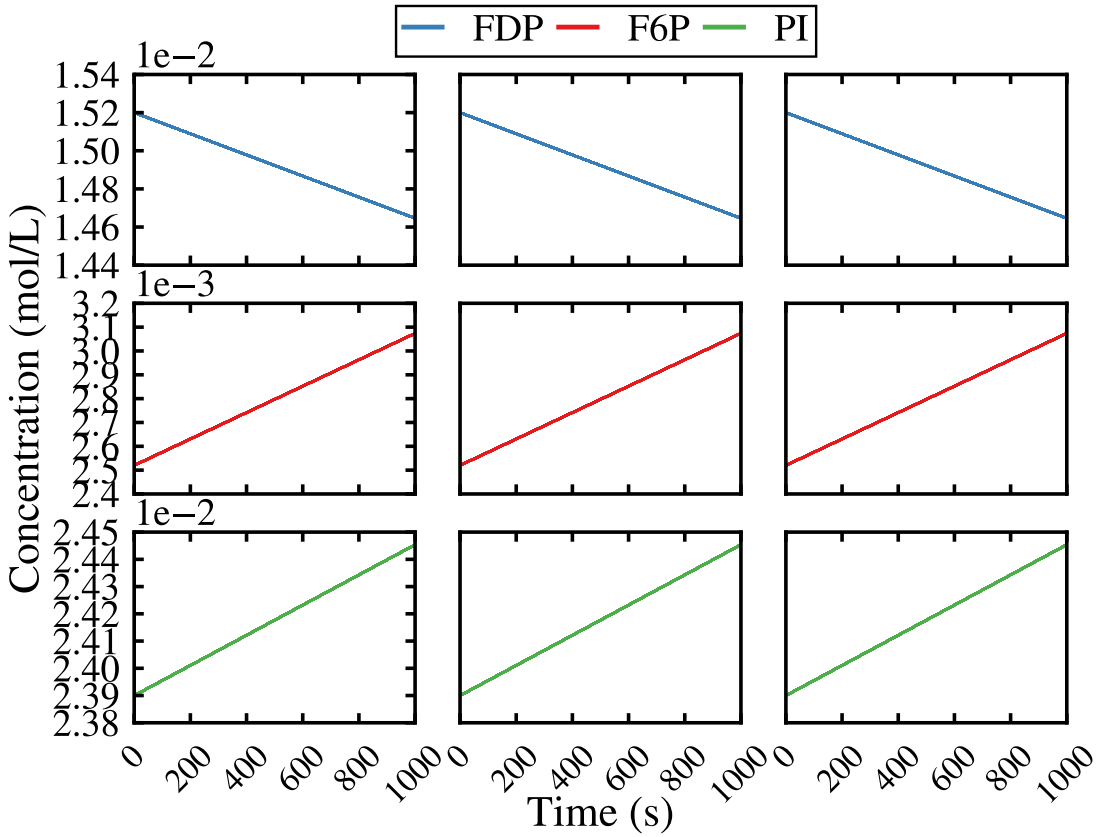


Figure 2.15: Metabolite time-courses obtained from simulated the irreversible Hill models. Each column represents a model ensemble, build from the parameters inferred through MASSef. Each row represents a different metabolite. For given metabolite, all models lead to the same predicted time-course.

2.4.8 Omics data used in FBA1 enzyme models

Species	Concentration (M)
FBA1	1.12×10^{-6} [36]
F6P	2.52×10^{-3} [29]
FDP	1.52×10^{-2} [29]
PI	2.39×10^{-2} [29]

Table 2.2: Omics data used to build the enzyme-level models for FBA1.

2.4.9 Omics data used in FBP2 enzyme models

Species	Concentration (M)
FBP2	9.72×10^{-8} [36]
DHAP	3.06×10^{-3} [29]
FDP	1.52×10^{-2} [29]
G3P	2.71×10^{-4} [29]

Table 2.3: Omics data used to build the enzyme-level models for FBP2.

References

[1] R. Adams, A. Clark, A. Yamaguchi, N. Hanlon, N. Tsorman, S. Ali, G. Lebedeva, A. Goltsov, A. Sorokin, O. E. Akman, C. Troein, A. J. Millar, I. Goryanin, and S. Gilmore. SBSI: An extensible distributed software infrastructure for parameter estimation in systems biology. *Bioinformatics*, 29(5):664–665, 2013.

[2] R. Alberty and G. G. Hammes. Application of the theory of diffusion-controlled reactions to enzyme kinetics. *The journal of physical chemistry*, 62(2):154–159, 1958.

[3] R. A. Alberty. The relationship between michaelis constants, maximum velocities and the equilibrium constant for an enzyme-catalyzed reaction. eng. *Journal of the american chemical society*, 75(8):1928–1932, 1953.

[4] S. Andreozzi, A. Chakrabarti, K. C. Soh, A. Burgard, T. H. Yang, S. Van Dien, L. Miskovic, and V. Hatzimanikatis. Identification of metabolic engineering targets for the enhancement of 1,4-butanediol production in recombinant *E. coli* using large-scale kinetic models. *Metabolic engineering*, 35:148–159, 2016.

[5] B. D. Bennett, E. H. Kimball, M. Gao, R. Osterhout, S. J. Van Dien, and J. D. Rabinowitz. Absolute metabolite concentrations and implied enzyme active site occupancy in *Escherichia coli*. *Nature chemical biology*, 5(8):593–9, Aug. 2009.

[6] C. Chassagnole, N. Noisommit-Rizzi, J. W. Schmid, K. Mauch, and M. Reuss. Dynamic modeling of the central carbon metabolism of *Escherichia coli*. *Biotechnology and bioengineering*, 79(1):53–73, 2002.

[7] A. Chowdhury, A. Khodayari, and C. D. Maranas. Improving prediction fidelity of cellular metabolism with kinetic descriptions. *Current opinion in biotechnology*, 36:57–64, 2015.

[8] W. W. CLELAND. The kinetics of enzyme-catalyzed reactions with two or more substrates or products. i. nomenclature and rate equations. eng. *Biochimica et biophysica acta*, 67:104–37, 104–137, 1963. ISSN: 18782434, 00063002.

[9] R. S. Costa, A. Veríssimo, and S. Vinga. KiMoSys : a web-based repository of experimental data for KInetic MOdels of biological SYStems. *Bmc systems biology*, 8:85, 2014.

[10] D. Davidi, E. Noor, W. Liebermeister, A. Bar-Even, A. Flamholz, K. Tummler, U. Barenholz, M. Goldenfeld, T. Shlomi, and R. Milo. Global characterization of *in vivo* enzyme catalytic rates and their correspondence to *in vitro* k_{cat} measurements. *Proceedings of the national academy of sciences*, 113(12):3401–3406, 2016.

[11] M. Elias, G. Wieczorek, S. Rosenne, and D. S. Tawfik. The universality of enzymatic rate-temperature dependency. eng. *Trends in biochemical sciences*, 39(1):1–7, 2014. ISSN: 13624326, 09680004. DOI: 10.1016/j.tibs.2013.11.001.

[12] K. v. Eunen and B. M. Bakker. The importance and challenges of *in vivo*-like enzyme kinetics. *Perspectives in science*, 1(1-6):126–130, 2014.

[13] A. Flamholz, E. Noor, A. Bar-Even, and R. Milo. EQuilibrator - The biochemical thermodynamics calculator. *Nucleic acids research*, 40(D1):770–775, 2012.

[14] A. Gábor and J. R. Banga. Robust and efficient parameter estimation in dynamic models of biological systems. *Bmc systems biology*, 9:74, 2015.

[15] A. Gábor, A. F. Villaverde, and J. R. Banga. Parameter identifiability analysis and visualization in large-scale kinetic models of biosystems. *Bmc systems biology*, 1154(11):1–16, 2017.

[16] R. García-Contreras, P. Vos, H. V. Westerhoff, and F. C. Boogerd. Why *in vivo* may not equal *in vitro* - New effectors revealed by measurement of enzymatic activities under the same *in vivo*-like assay conditions. *Febs journal*, 279(22):4145–4159, 2012.

[17] A. J. Hanekom. Generic kinetic equations for modelling multisubstrate reactions in computational systems biology. PhD thesis. University of Stellenbosch, 2006. URL: http://hdl.handle.net/10019.1/1659.

[18] J. J. Heijnen and P. J. T. Verheijen. Parameter identification of *in vivo* kinetic models: Limitations and challenges. *Biotechnology journal*, 8(7):768–775, 2013.

[19] S. Hoops, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI - A COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, 2006.

[20] N. Ishii, Y. Suga, A. Hagiya, H. Watanabe, H. Mori, M. Yoshino, and M. Tomita. Dynamic simulation of an *in vitro* multi-enzyme system. *Febs letters*, 581(3):413–420, 2007.

[21] N. Jamshidi and B. Ø. Palsson. Mass action stoichiometric simulation models: incorporating kinetics and regulation into stoichiometric models. *Biophysical journal*, 98(2):175–85, Jan. 2010.

[22] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: open source scientific tools for Python. 2001–. URL: `http://www.scipy.org/`.

[23] A. Khodayari and C. D. Maranas. A genome-scale escherichia coli kinetic metabolic model satisfying flux data for multiple mutant strains. *Nature communications*, 7:1–12, 2016. ISSN: 2041-1723. DOI: `10.1038/ncomms13806`. URL: `http://dx.doi.org/10.1038/ncomms13806`.

[24] O. Kotte and M. Heinemann. A divide-and-conquer approach to analyze underdetermined biochemical models. *Bioinformatics*, 25(4):519–525, 2009.

[25] H. Lineweaver and D. Burk. The determination of enzyme dissociation constants. eng. *Journal of the american chemical society*, 56:658–666, 1934. ISSN: 15205126, 00027863.

[26] H. Link, K. Kochanowski, and U. Sauer. Systematic identification of allosteric protein-metabolite interactions that control enzyme activity *in vivo*. *Nature biotechnology*, 31(4), 2013.

[27] P. Millard, K. Smallbone, and P. Mendes. Metabolic regulation is sufficient for global and robust coordination of glucose uptake, catabolism, energy production and growth in *Escherichia coli*. *Plos computational biology*, 13(2):e1005396, 2017.

[28] L. Miskovic and V. Hatzimanikatis. Production of biofuels and biochemicals: In need of an ORACLE. *Trends in biotechnology*, 28(8):391–397, 2010.

[29] J. O. Park, S. A. Rubin, Y.-f. Xu, D. Amador-noguez, J. Fan, T. Shlomi, and J. D. Rabinowitz. Metabolite concentrations, fluxes and free energies imply efficient enzyme usage. *Nature chemical biology*, (May), 2016.

[30] S. Placzek, I. Schomburg, A. Chang, L. Jeske, M. Ulbrich, J. Tillack, and D. Schomburg. BRENDA in 2017: New perspectives and new tools in BRENDA. *Nucleic acids research*, 45(D1):D380–D388, 2017.

[31] B. A. Reyes, J. S. Pendergast, and S. Yamazaki. Mammalian Peripheral Circadian Oscillators Are Temperature Compensated. *Journal of biological rhythms*, 23(1):95–98, 2008.

[32] P. A. Saa and L. K. Nielsen. Formulation, construction and analysis of kinetic models of metabolism: A review of modelling frameworks. *Biotechnology advances*, 2017.

[33] P. Saa and L. K. Nielsen. A general framework for thermodynamically consistent parameterization and efficient sampling of enzymatic reactions. *Plos computational biology*, 11(4):e1004195, 2015.

[34] E. Sanchez de Jimenez, E. Lee, J. Torres, and G. Soberon. On the mechanism of the effect of ionic strength on crystalline aldolase activity. eng. *Journal of biological chemistry*, 239:4154–4158, 1964.

[35] G. Savoglidis, A. X. da Silveira dos Santos, I. Riezman, P. Angelino, H. Riezman, and V. Hatzimanikatis. A method for analysis and design of metabolism using metabolomics data and kinetic models: Application on lipidomics using a novel kinetic model of sphingolipid metabolism. *Metabolic engineering*, 37:46–62, 2016.

[36] A. Schmidt, K. Kochanowski, S. Vedelaar, E. Ahrne, B. Volkmer, L. Callipo, K. Knoops, M. Bauer, R. Aebersold, and M. Heinemann. The quantitative and condition-dependent *Escherichia coli* proteome. *Nature biotechnology*, in press(December), 2015.

[37] K. Smallbone, H. L. Messiha, K. M. Carroll, C. L. Winder, N. Malys, W. B. Dunn, E. Murabito, N. Swainston, J. O. Dada, F. Khan, P. Pir, E. Simeonidis, I. Spasić, J. Wishart, D. Weichart, N. W. Hayes, D. Jameson, D. S. Broomhead, S. G. Oliver, S. J. Gaskell, J. E. G. McCarthy, N. W. Paton, H. V. Westerhoff, D. B. Kell, and P. Mendes. A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *Febs letters*, 587(17):2832–41, Sept. 2013.

[38] S. Srinivasan, W. R. Cluett, and R. Mahadevan. Constructing kinetic models of metabolism at genome-scales: A review. *Biotechnology journal*, 10(9):1345–1359, 2015.

[39] B. Teusink, J. Passarge, C. Reijenga, E. Esgalhado, C. v. d. Weijden, M. Schepper, M. Walsh, B. Bakker, K. v. Dam, H. Westerhoff, and J. Snoep. Can yeast glycolysis be understood in terms of *in vitro* kinetics of the constituent enzymes? testing biochemistry. eng. *European journal of biochemistry*, 267(17):5313–5329, 2000.

[40] K. F. Tipton and H. B. F. Dixon. Effects of pH on enzymes. *Methods in enzymology*, 63(63):183–234, 1979.

[41] L. M. Tran, M. L. Rizk, and J. C. Liao. Ensemble modeling of metabolic networks. *Biophysical journal*, 95(12):5606–17, Dec. 2008.

[42] K. Van Eunen, J. Bouwman, P. Daran-Lapujade, J. Postmus, A. B. Canelas, F. I. C. Mensonides, R. Orij, I. Tuzun, J. Van Den Brink, G. J. Smits, W. M. Van Gulik, S. Brul, J. J. Heijnen, J. H. De Winde, M. J. Teixeira De Mattos, C. Kettner, J. Nielsen, H. V. Westerhoff, and B. M. Bakker. Measuring enzyme activities under standardized in *vivo*-like conditions for systems biology. *Febs journal*, 277(3):749–760, 2010.

[43] J. T. Yurkovich, D. C. Zielinski, L. Yang, G. Paglia, O. Rolfsson, O. E. Sigurjonsson, J. T. Broddrick, A. Bordbar, K. Wichuk, S. Brynjolfsson, S. Palsson, S. Gudmundsson, and B. O. Palsson. Quantitative time-course metabolomics in human red blood cells reveal the temperature dependence of human metabolic networks. eng. *Journal of biological chemistry*, 292(48):19556–19564, 2017. ISSN: 1083351x, 00219258, 10678816. DOI: 10.1074/jbc.M117.804914.

Chapter 3

Elementary Mass Action Stoichiometric Simulation models predict non-negligible fractions of enzyme-bound metabolite concentrations

Abstract

To improve cell factory performance to the point where production of chosen molecules becomes economically profitable, extensive metabolic modification is required. Computational metabolic models have been used to both provide new insights into the inner workings of metabolism and new directions for strain engineering. Kinetic models in particular are key to model the dynamics of metabolism and substrate-level enzyme regulation. Two main approaches to build kinetic models are: bottom-up, which relies mostly on enzyme kinetic data, and top-down which relies mostly on omics data. Here, we use the elementary Mass Action Stoichiometric Simulation (eMASS) framework, an approach that integrates both enzyme kinetic data and omics data, to build a prototype kinetic model ensemble for eight enzymes in *E. coli*'s glycolysis. Following eMASS we decompose each reaction into elementary steps, so that we explicitly model free and enzyme-bound metabolite concentrations. This way we can drop the assumption that enzyme-bound metabolite concentrations are negligible, i.e. $x_{tot} \approx x_{free}$. Preliminary results show that the fraction of enzyme-bound metabolite vs. total metabolite can be as high as 0.4, i.e. the amount of enzyme-bound metabolite may not be negligible.

The underlying code is available at https://github.com/martamatos/eMASS.

3.1 Introduction

Cell factories have the potential to provide a more sustainable alternative to produce many of the molecules our society relies on. However, achieving yields that are commercially viable typically requires extensive cell engineering. Well studied organisms such as *Escherichia coli* and *Saccharomyces cerevisiae* are commonly used in cell factories due to the amount of existing tools to genetically engineer them. Since these are well studied organisms, several computational models for *E. coli*'s and *S. cerevisiae*'s metabolism have been built throughout the years, from simple stoichiometric models at genome-scale [12, 24, 21, 10] to generally smaller kinetic models [16, 17, 19, 20, 31, 7, 26, 14, 9], and some combinations thereof [8, 29], as well as models integrating metabolic networks with signaling networks [6], gene networks [23], or including enzyme structural details [4].

While stoichiometric models have provided new insights over the years on the inner workings of metabolism [18], these do not account for substrate-level enzyme regulation, which can be key to, e.g. make sense of omics data. For instance, van Eunen *et al* [9] showed that, if pyruvate kinase activation by fructose-1,6-biphosphate (F16bP) is not included in a *S. cerevisiae* glycolysis model, there is accumulation of 3-phospho-D-glycerate (3PG), 2-phospho-D-glycerate (2PG), and phosphoenolpyruvate (PEP) due to a lack of homeostatic regulation. To explicitly model substrate-level enzyme regulation we need to resort to kinetic models which are able to provide detailed information on time-dependent metabolite concentrations and reaction fluxes. However, this extra information comes at the cost of more data to parameterize the models. To build a metabolic kinetic model, two main approaches can be distinguished: a bottom-up approach and a top-down approach. Briefly, in a bottom-up approach the rate laws that model each enzymatic reaction are parameterized individually by, e.g. measuring the enzyme's macroscopic kinetic parameters k_{cat}, K_m; while in a top-down approach the rate laws are formulated and the kinetic parameters are determined by fitting the rate law equations to omics data. However, it is not uncommon to start with a bottom-up approach and fit unknown parameters at the system level [19].

Here, we start by following a bottom-up approach, where we first model each enzyme individually by generating fully functional enzyme-level kinetic Ordinary Differential Equations (ODE) models. Each differential equation models the concentration changes for a given metabolite or enzyme form, where enzyme forms refer to both free

enzyme and enzyme bound to substrate(s)/product(s). We then combine the enzyme-level models to assemble a prototype ensemble of system-level ODE models for eight enzymes in *E. coli*'s glycolysis. We call this approach to assemble a system-level ODE model out of individual enzyme-level ODE models eMASS, elementary Mass Action Stoichiometric Simulation. eMASS builds on the MASS formalism [15] but decomposes all enzymatic reactions into their respective elementary reactions. While in the original MASS approach only regulatory enzymatic reactions are decomposed into elementary steps and the remaining reactions are modeled by generalized mass action rates law which are parameterized by integrating omics data and equilibrium constants. eMASS was implemented as a software package, MASSef, and introduced in chapter 2. As in other current approaches for building kinetic models that distinguish different enzyme forms (i.e. complexes of enzyme bound to substrates/products and free enzyme) [32, 17, 28, 27, 20, 19], in eMASS we initially assumed the concentration of metabolites bound to enzymes to be negligible, i.e. $x_{free} \approx x_{tot}$, where x_{tot} is the total metabolite concentration in the cell and x_{free} is the concentration of metabolite not bound to any macromolecule. Total metabolite concentrations x_{tot} are commonly reported in metabolic profiling data obtained through mass spectrometry, as these experimental approaches cannot distinguish between metabolite free in the cell and bound to macromolecules [3, 2, 16, 5]. However, preliminary results from building a kinetic model for each enzyme individually showed that the assumption $x_{free} \approx x_{tot}$ may not always hold. Therefore, we drop the assumption $x_{free} \approx x_{tot}$ and explicitly model both concentrations.

When distinguishing explicitly free and total metabolite concentrations, x_{free} is the quantity being simulated in the model while x_{tot} only defines the value taken by the sum of free metabolite and metabolite bound to enzyme(s). However, we only get x_{tot} from metabolomics experiments and thus need to determine the initial concentrations for $x_{free}$to be able to simulate the ODE model. Consequently, when we fully build each enzyme-level ODE model individually and then combine these models into an ODE system-level model, we can no longer guarantee that the initial concentrations of free metabolite are non-negative. For instance, if a given metabolite A participates in two distinct reactions, and in the enzyme model for reaction r_1 70% of A is predicted to be bound to enzyme at $t = 0$ s and in the enzyme model for reaction r_2 50% of the same metabolite A is predicted to be bound to the enzyme at $t = 0$ s, when we put these models together to build a system-level ODE model, 120% of the total metabolite concentration will be bound to enzyme at $t = 0$ s, and the initial free metabolite concentration will be -20% of the total initial metabolite concentration. Hence, to

avoid negative initial free metabolite concentrations,we develop a two-step approach, eMASS2, to determine the initial free metabolite concentrations and thus set up an ODE system-level model. In eMASS2, we first use MASSef to generate a set of rate constants for each enzymatic reaction, and then build a system of equations at the system-level composed of all steady-state flux equations for the enzymes in the system, and a total metabolite conservation equation for each metabolite in the system, where $x_{tot} = \sum_i x_{bound,i} + x_{free}$. Both the flux and the metabolite conservation equations depend on steady-state concentrations of total enzyme and free metabolite, which are then sampled to fit steady-state fluxomics and metabolomics data at the system level. The sampled free metabolite concentrations and total enzyme concentrations are then defined as the initial concentrations in the system-level ODE model.

We use both eMASS and eMASS2 to build a model prototype that includes eight enzymes in *Escherichia coli*'s glycolysis: FBA1, FBA2, TPI, GAPDH, PGK, PGMd, PGMi, and ENO. In eMASS, a model is built for each enzyme individually, using MASSef, and the system-level model is defined as the union of all enzyme-level models, while in eMASS2, MASSef is used to generate rate constant sets for each enzyme, and enzyme concentrations as well as free metabolite concentrations are globally fit to fluxomics and metabolomics data. Preliminary results show that, although in many cases the amount of metabolite bound to enzymes can indeed be neglected, in some cases it may go up to 40% of the total metabolite concentration.

3.2 Methods

3.2.1 Enzyme-level models

Using MASSef (see chapter 2 for more details), we build an enzyme-level ODE model ensemble for each enzyme in our system-level ODE model prototype. To build each enzyme-level model we follow the workflow in Figure 3.1. First, we decompose the reaction into its elementary steps, according to the enzyme mechanism, and add regulatory steps if needed. Second, we generate a mass-action rate law for each elementary reaction, which results in two elementary rate constants per elementary reaction, $\overrightarrow{k}$ and $\overleftarrow{k}$ describing the reaction velocities of the forward and reverse directions, respectively. Third, we find the overall steady-state flux equation, v_{ss}, by solving the system of mass balance equations at steady-state subject to the conservation of total enzyme concentration constraint, i.e. the total enzyme concentration is the sum of its enzyme forms (enzyme bound to substrates or products and free enzyme). Assuming the net

flux through the reaction's conversion step is equal to the reaction's measured flux, this leads to the overall flux equation, v_{ss}, for each enzymatic reaction as a function of rate constants $\boldsymbol{k}$, metabolite concentrations $\boldsymbol{x}$, and total enzyme concentration E_{tot}. This flux equation v_{ss} is then used as a basis to generate the equations used to fit the kinetic data (e.g. K_m, k_{cat}, K_i). Here, we use steady-state equations to fit the kinetic data because the steady-state was also assumed when measuring such data. This assumption is valid as long as the substrate concentration is much larger than the enzyme concentration, which is typically the case in *in vitro* experiments. Once we obtain the rate constant values, these are valid for other conditions as well.

To illustrate the workflow we take the reaction catalyzed by Enolase as an example, a one substrate and one product reaction in glycolysis. We start by decomposing the reaction into its elementary steps:

$$ENO + 2PG \rightleftharpoons ENO\&2PG \rightleftharpoons ENO\&PEP \rightleftharpoons ENO + PEP$$

where $ENO\&2pg$ is the complex formed by the enzyme and the substrate 2PG. We generate a mass action rate law for each step:

$$\begin{aligned} v_1 &= \overrightarrow{k}_1[ENO][2PG] - \overleftarrow{k}_1[ENO\&2PG] \\ v_2 &= \overrightarrow{k}_2[ENO\&2PG] - \overleftarrow{k}_2[ENO\&PEP] \\ v_3 &= \overrightarrow{k}_3[ENO\&PEP] - \overleftarrow{k}_3[ENO][PEP] \end{aligned}$$

we assume that the reaction's steady-state flux is the same as the flux through the conversion step v_2:

$$v_{ss} = v_2 = \overrightarrow{k}_2[ENO\&2PG] - \overleftarrow{k}_2[ENO\&PEP] \tag{3.1}$$

to find $[ENO\&2PG]$ and $[ENO\&PEP]$ in terms of rate constants $\boldsymbol{k}$ and metabolite concentrations $[2PG]$ and $[PEP]$, we solve the system of steady-state equations for all enzyme-form concentrations together with the total enzyme conservation equation :

$$\begin{aligned}
\frac{d}{dt}[ENO] &= v_3 - v_1 = 0 \\
\frac{d}{dt}[ENO\&2PG] &= v_1 - v_2 = 0 \\
\frac{d}{dt}[ENO\&PEP] &= v_2 - v_3 = 0 \\
ENO_{tot} &= [ENO] + [ENO\&2PG] + [ENO\&PEP]
\end{aligned}$$

once this system is solved and we know $[ENO\&2PG] = h_{[ENO\&2PG]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$ and $[ENO\&PEP] = h_{[ENO\&PEP]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$, we substitute these into equation 3.1 and get $v_{ss} = f_{v_{ss}}(ENO, 2PG, PEP, \boldsymbol{k})$:

$$v_{ss} = f_{v_{ss}}(ENO, 2PG, PEP, \boldsymbol{k})$$

$$= \overrightarrow{k}_2 \cdot h_{[ENO\&2pg]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k}) - \overleftarrow{k}_2 \cdot h_{[ENO\&pep]}(ENO_{tot}, 2PG, PEP, \boldsymbol{k})$$

$$= \frac{[ENO_{tot}]([2PG]\overrightarrow{k}_1\overrightarrow{k}_2\overrightarrow{k}_3 - [PEP]\overleftarrow{k}_1\overleftarrow{k}_2\overleftarrow{k}_3)}{[2PG]\overrightarrow{k}_1(\overrightarrow{k}_2 + \overleftarrow{k}_3 + \overrightarrow{k}_3) + \overleftarrow{k}_1(\overrightarrow{k}_2 + \overleftarrow{k}_3) + \overrightarrow{k}_2\overrightarrow{k}_3 + [PEP]\overleftarrow{k}_2(\overleftarrow{k}_1 + \overleftarrow{k}_3 + \overrightarrow{k}_3)}$$

We then use v_{ss} to generate the appropriate equations to fit the elementary rate constants $\boldsymbol{k}$ to kinetic data (k_{cat}, K_m, K_i, etc.) while assuming $E_{tot,i}$ to be 1. In particular, k_{cat} values are defined as an effective k_{cat} and are simply v_{ss}/E_{tot} when product concentrations are null. K_m is also defined as an effective K_m and is simply $v_{ss}/V_{max,eff} = 0.5$ solved for the substrate in question. $V_{max,eff}$ is the effective V_{max}, i.e. v_{ss} when product concentrations are null and the concentration for the substrate in question is saturating. We assume a saturating concentration of 1 M. Since in general the kinetic data can be explained equally well by different sets of elementary rate constants, this leads not to a single model, but to a model ensemble, where each model corresponds to a different set of rate constants $\boldsymbol{k}$. For further details see chapter 2.

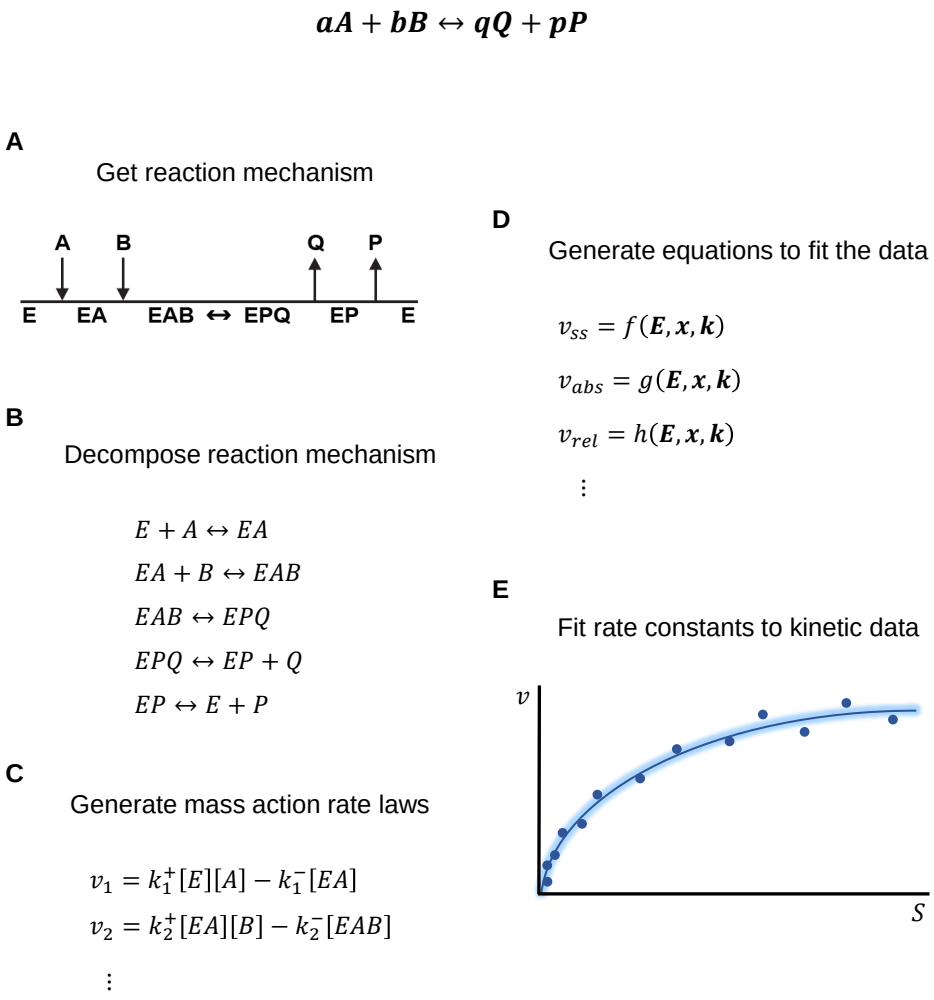


Figure 3.1: MASSef workflow to generate rate constant sets that fit kinetic data. For a given reaction (A) get the reaction mechanism from literature, (B) decompose the reaction mechanism into its elementary steps, (C), generate a mass action rate law for each elementary step, (D) generate flux equation and equations to fit the kinetic data (k_{cat}, K_m, etc), (E) fit the rate constants to kinetic data using PSO (particle swarm optimization) and LMA (Levenberg-Marquardt algorithm).

3.2.2 System-level model

eMASS

To set up a system-level ODE model that can be simulated we take the rate constants determined by MASSef for each enzyme, but we still need to define the initial concentrations for free metabolite and enzyme forms. Thus, we follow the workflow in

Figure 3.2, where, for each enzyme-level model, we start by finding all enzyme form concentrations. Hence, we:

1. take $v_{ss,i} = f_{v_{ss,i}}(E_{tot,i}, \boldsymbol{x}, \mathbf{k})$ (defined above), where i represents reaction i in a system of n reactions;

2. calculate the steady-state total enzyme concentration in terms of $v_{ss,i}$: $E_{tot,i} = g_{E_{tot,i}}(v_{ss_i}, \boldsymbol{x}_{tot}, \mathbf{k})$, by solving $f_{v_{ss,i}}(E_{tot,i}, \boldsymbol{x}, \mathbf{k})$ for $E_{tot,i}$ and integrate the rate constant values, experimental steady-state total metabolite concentrations, $\boldsymbol{x}_{tot}$, and steady state flux for each reaction, $v_{ss,i}$, obtained either from fluxomics data or Flux Balance Analysis (FBA) (or similar methods);

3. calculate each enzyme form concentration e_{ij} by solving $h_{e_{ij}}(E_{tot,i}, \boldsymbol{x}_{tot}, \mathbf{k})$, where j represents enzyme form j of enzyme i.

The calculated enzyme form concentrations are then integrated in the enzyme-level model as initial concentrations for the enzyme forms. This results in an enzyme-level model that can be simulated, and by taking the union of all enzyme-level ODE models we get a working system-level ODE model. Finally, in the system-level model we update the free metabolite concentrations x_{free} by subtracting the enzyme-bound metabolite concentrations x_{bound} from x_{tot}. However, because we do not have only one model for each enzyme but a model ensemble for each enzyme, we take all the combinations of the best three models per enzyme (lowest sum of squared errors) to build a system-level model ensemble comprised of 6561 models.

Alternatively, the model can be built by integrating steady-state total enzyme concentrations $E_{tot,i}$ instead of steady-state reaction fluxes v_{ss}. For a detailed example, see chapter 2.

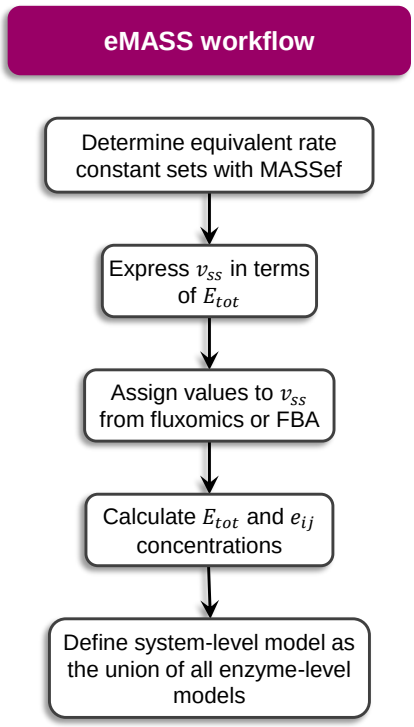


Figure 3.2: eMASS workflow to build system-level models by combining fully functional enzyme-level kinetic models.

eMASS2

A more precise way to determine the initial free metabolite and enzyme form concentrations is to use eMASS2. To set up a system-level ODE model using eMASS2, we follow the workflow in Figure 3.3. First, we take all the combinations of the best three enzyme-level ODE models for each enzyme which, for eight enzymes, leads to 6561 system-level ODE models. For each model with n reactions ($n = 8$ in this case) and m_1 metabolites ($m_1 = 13$ in this case) we define the following system of equations:

$$\begin{aligned} v_{ss_i} &= f_{v_{ss,i}}(E_{tot_i}, x_{jl}, \mathbf{k}), \\ x_{tot_j} &= \sum_{l=1}^{m_2} x_{jl} \end{aligned}$$

where $i \in [1, n]$, $j \in [1, m_1]$ and j represents metabolite j, and $l \in [1, m_2]$ represents the metabolite form l of metabolite j which can be found in m_2 forms (enzyme-bound and free forms). The sum of both free and enzyme-bound metabolites x_{jl} must

be equal to the respective steady-state total concentration, x_{tot_j}, measured experimentally. The steady-state flux equation $f_{v_{ss,i}}(E_{tot_i}, x_{jl}, \mathbf{k})$, defined for each reaction (the same equation as in chapter 2) must fit the steady-state fluxes, v_{ss_i}, measured either experimentally or obtained from FBA or similar methods. We use the Levenberg-Marquardt algorithm (implemented in the LMfit package for Python [22]) to solve this system of equations, by finding free metabolite concentrations x_{jl} and steady-state total enzyme concentrations E_{tot_i} that minimize the differences between 1) the measured v_{ss_i} and the calculated $f_{v_{ss,i}}(E_{tot,i}, x_{jl}, \mathbf{k})$, and 2) the measured x_{tot_j} and the sum x_{jl} over all l metabolite forms. We consider only models where the difference between the true and predicted v_{ss_i} and x_{tot_j} values is less than 10%, and discard the others. The enzyme form concentrations are calculated from $h_{[enz-form]}(E_{tot}, \boldsymbol{x}_{free}, \boldsymbol{k})$ defined in section 3.2.1. The resulting steady-state free metabolite and enzyme form concentrations are then defined as initial concentrations in the respective system-level ODE model. Note that these steady-state flux equations are used only to find the initial steady-state free metabolite and enzyme form concentrations for the respective system-level ODE model. The reason we use steady-state flux equations to do so is because: 1) to the best of our knowledge, most omics datasets are measured at steady-state; and 2) it is an efficient way to find the values for free metabolite and total enzyme initial concentrations.

For isoenzymes we set the flux through the respective enzyme as the ratio of the isoenzyme concentration and the sum of all isoenzyme concentrations, e.g. if [FBA1]=1×10^{-6} M and [FBA2]=3×10^{-6} mol/L, then the flux through FBA1 will be 1/4 of the total, while the flux through FBA2 will be 3/4 of the total.

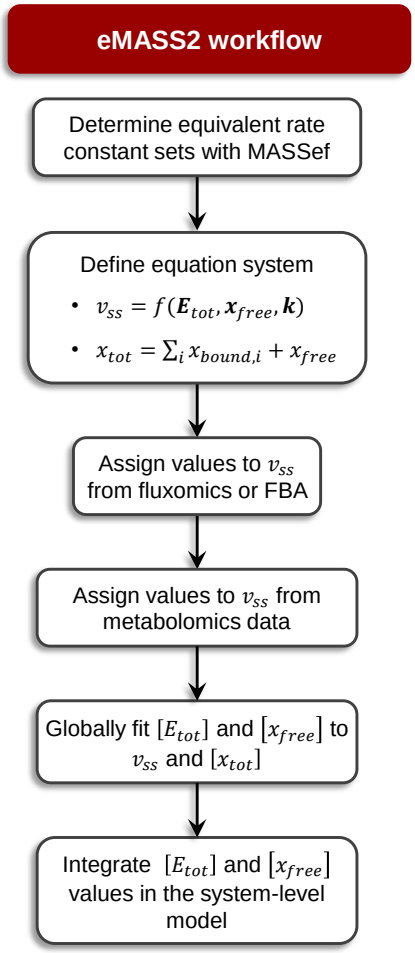


Figure 3.3: eMASS2 workflow to build system-level kinetic models by integrating fluxomics and metabolomics data and globally sampling total enzyme and free metabolite concentrations.

Total enzyme concentrations were sampled in the interval $[10^{-12}, 5 \times 10^{-3}]$ M and free metabolite concentrations were sampled in the interval $[10^{-12}, 10^{-1}]$ M.

Alternatively, total enzyme concentrations $E_{tot,i}$ can be expressed in terms of $v_{ss,i}$, and proteomics data can be integrated instead of fluxomics.

3.2.3 Data used for model fitting

The kinetic data used to fit each enzyme model individually can be found here. The metabolomics data was obtained from [25], where *E. coli* K-12 strain NCM3722 was grown in Gutnick minimal medium [11]. The proteomics data was obtained from [30] for *E. coli* BW25113 grown in glucose. And the fluxomics data was obtained from parsimonious Flux Balance Analysis of the iJO1366 model [24], with the glucose and the oxygen flux set to -2.93 mmol/gDW/h and -5.79 mmol/gDW/h, respectively.

The objective function for pFBA is cell growth. The results were then converted to mol/L/s, using a dry cell weight of 2.8×10^{-13} gDW from [13], and cell volume of 3.2×10^{-15} L from [33] for cells grown on glucose.

3.2.4 Model simulations

To check for differences in the dynamic behavior between the ODE models assembled through eMASS and the ODE models assembled through eMASS2, we simulate the models for a 100 s as an open system.

Both metabolite/enzyme concentrations and reaction fluxes at $t = 0$ s are steady-state concentrations/fluxes. Thus, we define exchange fluxes for each modeled metabolite such that, in the absence of perturbations, the models are effectively at steady-state at $t = 0$ s. Except for FBP and PEP, the exchange flux for each metabolite is constant and defined as the difference between the fluxes for reactions that produce and consume a given metabolite. For instance, 2PG is produced by PGM and consumed by ENO, thus the exchange flux for 2PG is $v_{ENO} - v_{PGM}$, where the exchange reaction is $2PG \rightarrow \emptyset$. For FBP the exchange flux is defined as $v_{EX,FBP} = \overrightarrow{k}_{EX,FBP}[FBP_{EX}]$, where $v_{EX,FBP}$ is the flux for the exchange reaction $FBP_{EX} \rightarrow FBP$ which is set to be equal to the flux through FBA at $t = 0$ s, the only reaction that consumes FBP. FBP_{EX} is the external concentration of FBP and is a constant defined to be the same as the intracellular concentration of free FBP at $t = 0$ s. Finally $\overrightarrow{k}_{EX,FBP} = {}^{v_{EX,FBP}}/_{[FBP_{EX}]}$. For PEP, the exchange flux is defined as $v_{EX,PEP} = \overrightarrow{k}_{EX,PEP}[PEP]$, where $v_{EX,PEP}$ is the flux for the exchange reaction $PEP \rightarrow \emptyset$, and is defined to be the same as the flux through ENO at $t = 0$ s, the only reaction that produces PEP. $[PEP]$ is the time-dependent intracellular concentration of free PEP. $\overrightarrow{k}_{EX,FBP}$ is defined based on the values for $v_{EX,PEP}$ and $[PEP]$ at $t = 0$ s as $\overrightarrow{k}_{EX,FBP} = {}^{v_{EX,PEP}}/_{[PEP]}$.

Finally, as we do not model co-factor regeneration, the concentrations for ATP, ADP, NAD, NADH, and PI are defined as constants, their values being set to their respective free metabolite concentrations predicted by eMASS/eMASS2.

3.3 Results

Using eMASS and eMASS2 we build two prototype model ensembles for eight enzymes in *E. coli*'s glycolysis: FBA1, FBA2, TPI, GAPDH, PGK, PGMd, PGMi, and ENO. We take all combinations of the best three enzyme-level models for each enzyme,

resulting in a system-level model ensemble with 6561 models. Note that we take only the best three enzyme-level models whose metabolite initial concentrations at the enzyme-level are non-negative. From here on, we refer to the model ensemble generated through eMASS as the eMASS model ensemble, and to the model ensemble generated through eMASS2 as the eMASS2 model ensemble. In the eMASS2 model ensemble only 3407 models are considered, as for the remaining models the difference between true and predicted fluxes and total metabolite concentrations is larger than 10%. For the eMASS and eMASS2 results to be comparable in a fair way, the eMASS model ensemble is reduced to the same 3407 models present in the eMASS2 model ensemble.

3.3.1 Predicted free metabolite concentrations

We start by comparing the fractions of bound metabolite concentrations predicted by each approach, see Figure 3.4. By fraction of bound metabolite we mean the concentrations ratio of metabolite bound to enzyme over total metabolite, $^{x_{bound}}/_{x_{tot}} = 1 - {}^{x_{free}}/_{x_{tot}}$. The results are relatively similar, with both approaches predicting non negligible fractions of bound metabolite. Both eMASS and eMASS2 predict non-negligible fractions of bound metabolite for 13DPG, 3PG, and PEP, while eMASS also predicts a ~ 0.15 fraction of bound metabolite for ATP. Yet, eMASS tends to predict higher fractions of bound metabolite than eMASS2. This is most likely due to the difference in the metabolite concentrations used to define the enzyme-form concentrations, which seems to lead to higher predictions for enzyme-bound metabolite concentrations. The enzyme form concentrations (free and metabolite bound enzyme) depend on the free metabolite concentrations. In eMASS the free metabolite concentrations used to calculate the enzyme-form concentrations are the total metabolite concentrations, which are equal or higher than the free metabolite concentrations used to calculate enzyme-form concentrations in eMASS2. Even though the relationship between enzyme-form concentration and free metabolite concentrations is non-linear, it is possible that higher free metabolite concentrations lead to higher metabolite-bound enzyme concentrations.

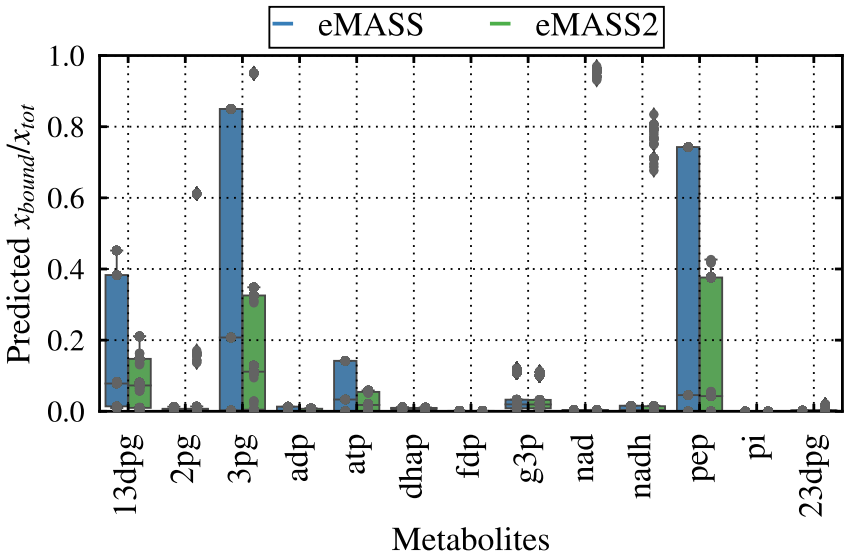


Figure 3.4: Predicted fraction of bound metabolite, x_{bound}, for each model in the eMASS (blue) and eMASS2 (green) model ensembles. Grey dots represent the particular values predicted by each approach.

The above results are not totally unexpected if we look at the total enzyme and total metabolite concentration distributions used to generate the models, which overlap slightly, see Figure 3.5. In this figure we considered only the sum of all isoenzymes for each reaction. The overlap is most likely due to the low concentrations for 13DPG, NADH, 2PG, and 23DPG, all with concentrations on the order of 10^{-5}M. For exact values see section 3.5. The metabolites with higher predicted fractions of bound metabolite have total concentrations ranging from 10^{-5} M for 13DPG to 10^{-3} M for 3PG, with PEP and ATP having a total concentration on the order of 10^{-4} M. What these metabolites have in common is that all are involved in reactions catalyzed by the enzymes with higher concentrations in the model, on the order of 10^{-5} M. This seems to indicate that higher total enzyme concentrations lead to higher fractions of bound metabolite, which intuitively makes sense. Yet, it also hints at another issue with this model ensemble prototypes: if for a given metabolite, only some of the reactions in which it is involved are included in the model (as opposed to all the reactions where it is either a product or substrate in the cell metabolism), the predicted free metabolite concentrations might be overestimated.

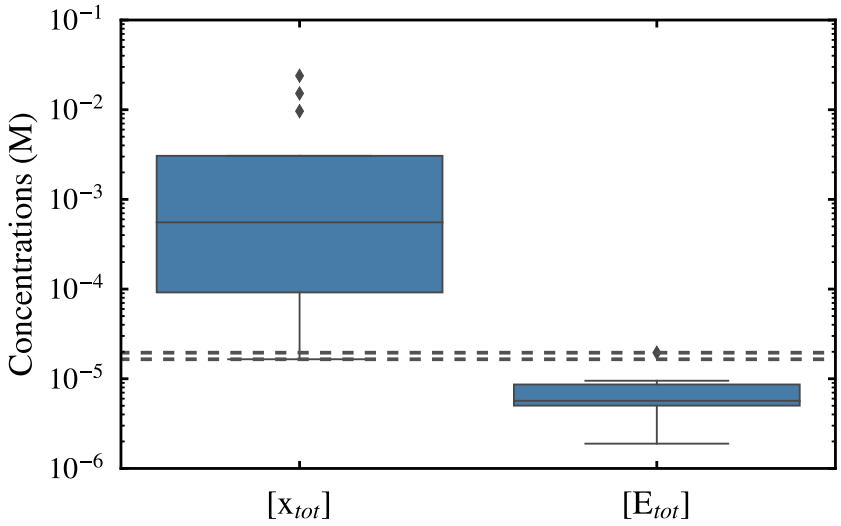


Figure 3.5: Distribution of total metabolite and total enzyme concentrations used to build the model ensembles. The space in between dashed lines corresponds to the intersection of both distributions.

Finally, note that all system-level models included both in the eMASS and the eMASS2 ensembles only include non negative free metabolite concentrations. However, for each enzyme we built between 50 and 100 enzyme-level models using the eMASS approach. Of these, the ones that do not fit the respective kinetic data with a tolerance of 10% are discarded. Still, some of the remaining enzyme-level models have negative initial free metabolite concentrations. In particular, for FBA1 one of the best three enzyme-level models contains a negative free metabolite concentration, and for PGK all three best models include a negative free metabolite concentrations. These are substituted by the next best models with non negative free metabolite concentrations. Yet, as a test, we also used eMASS2 to build the system-level model ensemble without substituting those enzyme-level models with negative initial free metabolite concentrations. It led to only ~ 700 models with a difference between true and predicted fluxes and total metabolite concentrations lower than 10%, compared to more than 3000 models when enzyme-level models with negative free metabolite concentrations were removed.

3.3.2 Predicted total enzyme concentrations

By integrating flux data, either experimentally measured or calculated through FBA methods, the total enzyme concentrations $E_{tot,i}$ are left as dependent variables that

can be calculated as a function of the elementary rate constants $\boldsymbol{k}$, the metabolite concentrations $\boldsymbol{x}$, and the flux values $v_{ss,i}$. Here we used flux data determined through pFBA. Thus, we look at the resulting total enzyme concentration values predicted by each model in the eMASS and eMASS2 ensembles, and compare both with experimental data from [30] and $^{v_i}/_{n*k_{cat,i}}$ in Figure 3.6. v is the flux obtained from pFBA, k_{cat} is the k_{cat} used to fit the enzyme rate constants, and n is the number of active sites in the enzyme. For the enzymes used in this model the number of enzyme subunits and active sites are the same. $E_{tot,i} = {}^{v_i}/_{n*k_{cat,i}}$ should provide a lower bound on the predicted total enzyme concentration, as it gives us the minimum enzyme concentration needed to catalyze a certain reaction with flux v_i . However, in practice it is likely that higher enzyme concentrations are needed to attain the same flux, as not all enzymes are necessarily in an active state [1] and substrate concentrations may not be saturating either. Also, for FBA1 the respective $^{v_i}/_{n*k_{cat,i}}$ is actually lower than the experimental $E_{tot,i}$, indicating that perhaps the k_{cat} used to fit the rate constants for this enzyme is too low. In fact, FBA1 has two sets of k_{cat} and K_m, one set measured with citrate as cosubstrate and another without citrate. When citrate is a co-substrate the k_{cat} value is higher. Yet, we used the k_{cat} determined in the absence of citrate, as citrate wasn't produced by any enzyme in the model.

Overall, eMASS models predict higher enzyme concentrations than eMASS2 models, predicting such high concentrations that are not even realistic. This might be again because in eMASS $E_{tot,i}$ is calculated based on total metabolite concentrations which are either equal or higher than the free metabolite concentrations. This can lead to higher enzyme-form concentrations and thus total enzyme concentrations. Regarding $E_{tot,i}$ predicted by eMASS2 models, these are always above the lower bound provided by $^{v_i}/_{kcat_i}$, making them consistent with the data used to fit the models. However, with a few exceptions, $E_{tot,i}$ values tend to be overestimated by eMASS2 models when comparing to the experimental $E_{tot,i}$ concentrations. This could be a result of not taking into account all the reactions in which the metabolites included in the models are either substrates or products, which most likely leads to higher free metabolite predictions than in a larger model.

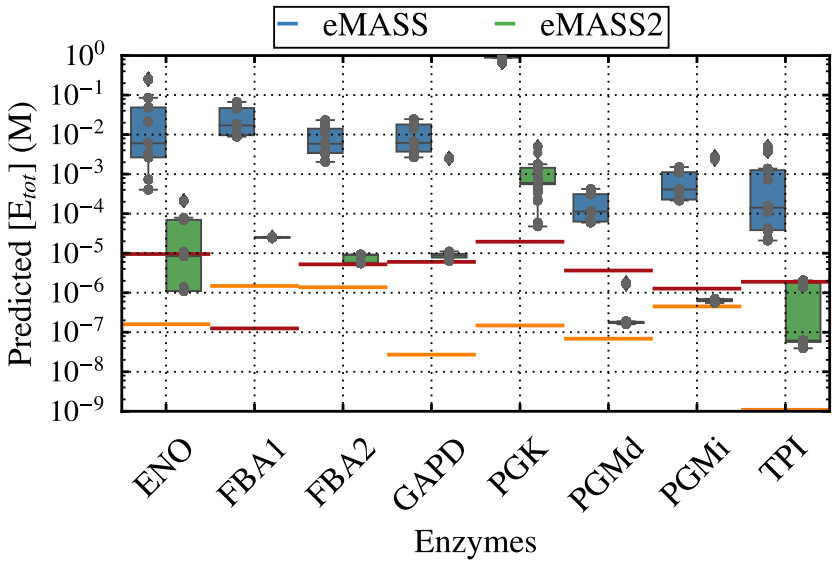


Figure 3.6: Predicted vs experimental total enzyme concentrations, E_{tot}. The boxplots represent the predicted enzyme concentrations by each model in each model ensemble: eMASS in blue, and eMASS2 in green. The red lines represent the measured enzyme concentrations in [30]. Grey dots represent the particular values predicted by each approach.

3.3.3 Comparing the dynamics of eMASS and eMASS2 model ensembles

Finally, we simulate all the 3407 models built both through eMASS and eMASS2 for 100 s. By visual inspection, we find that simulating the models for 100 s is enough to assess their behavior in the absence of perturbations, in particular if the models are at steady-state at $t = 0$ s and whether metabolite concentrations diverge or not. We do not compare their predictions to experimental time-course data, because we consider the models to be very incomplete at this point, missing key enzymes, such as PFK and PYK, which modulate the dynamic behavior of glycolysis. These models are simply a prototype and at this stage we want to compare only the eMASS and eMASS2 approaches.

The free metabolite time-courses resulting from simulating the eMASS and eMASS2 ensembles are shown in Fig. 3.7. While all free metabolite concentrations in the eMASS2 models (Fig. 3.7B) are constant for most of the simulation, these are definitely not constant in the eMASS models (Fig. 3.7A). Even though both model ensembles are fitted with steady-state data, the fluxes in eMASS2 are fitted at the

system-level and the free metabolite and total enzyme concentrations are such that the whole system will indeed be at steady-state, when the appropriate exchange reactions are included in the model. On the other hand, the fluxes in eMASS are fitted at the enzyme-level, which means total enzyme concentrations are determined at the enzyme level, such that the enzyme model is at steady-state at $t = 0$ s. But this does not mean that the system-level model will be at steady-state at $t = 0$ s once all enzymes are combined into a single system-level. Therefore, the system-level models built through eMASS will not be at steady-state at $t = 0$ s. However, these do seem to reach a steady-state within 100 s. Also, we should note that the metabolite concentrations generated through eMASS2 start diverging slightly after 50 s of simulation. This can have two reasons: 1) numerical precision in the calculation of the metabolite concentrations over time, and 2) some of the Jacobian eigenvalues have positive real parts, with the highest one being on the order to 10^{-3}.

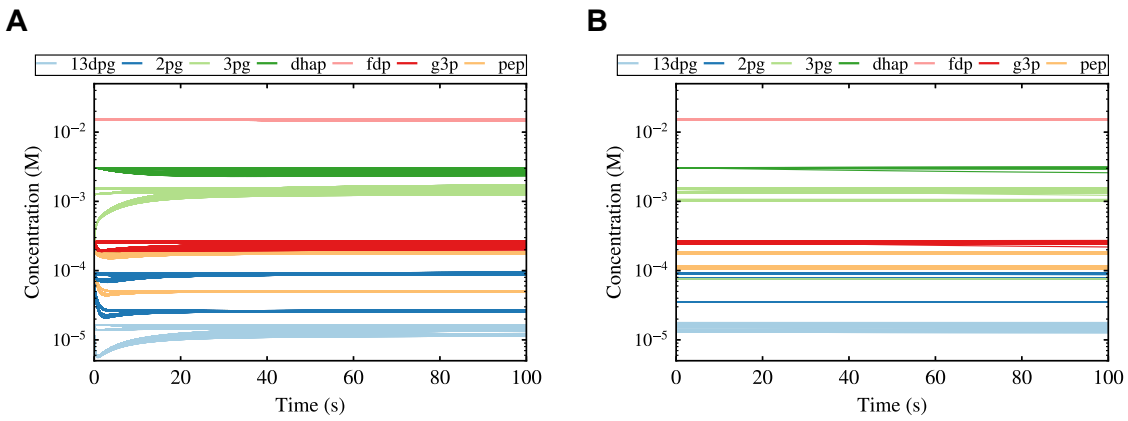


Figure 3.7: Free metabolite time courses for (A) eMASS and (B) eMASS models in the absence of perturbations.

3.4 Conclusions

Here, we presented and compared two new approaches to build system-level kinetic models: eMASS where we use MASSef to generate equivalent rate constant sets for each enzyme compatible with the experimental kinetic data, and integrate fluxomics and metabolomics data with each enzyme-level model individually, leading to a fully functional enzyme-level kinetic model. The system-level kinetic model is then defined as the union of all the enzyme-level models. In eMASS2 we use MASSef to generate

equivalent rate constant sets for each enzyme compatible with the experimental kinetic data, and then globally fit total enzyme and free metabolite concentrations to fluxomics and metabolomics data. One key difference between these methods is that enzyme concentrations in eMASS2 are a function of free metabolite concentrations, as the latter are predicted at the system level. On the other hand, in eMASS, because fluxes are integrated individually with each enzyme model, the free metabolite concentrations at the system-level are unknown. Thus, we use the total metabolite concentrations to calculate enzyme concentrations. This leads to predictions of enzyme concentrations higher than in eMASS2 models and to higher enzyme-bound metabolite concentrations. In eMASS free metabolite concentrations are calculated after assembling the system-level model by simply subtracting enzyme-bound metabolite concentrations from the total metabolite concentrations. This would also explain: 1)why eMASS tends to predict higher total enzyme concentrations and higher fractions of enzyme-bound metabolite, and 2) why the eMASS models are not at steady-state at $t = 0$ s in the absence of perturbations.

eMASS2 is no doubt a more correct approach to build the system-level kinetic model, as fluxes are functions of free metabolite concentrations, as it should be, and not total metabolite concentrations. Also it leads to considerably less overestimation of total enzyme concentrations, and the models generated through eMASS2 fitted with steady-state omics data are indeed at steady-state at $t = 0$ s in the absence of perturbations. The only reason that could eventually justify the use of eMASS instead of eMASS2, is that eMASS is easier to scale up to hundreds of enzymes as it does not involve global fitting. However, it would probably need to be re-thought and its predictions thoroughly validated.

Summing up, besides being a more correct approach to build a system-level kinetic model for metabolism, eMASS2 also leads to more reasonable results. It predicts non-negligible fractions of enzyme-bound metabolite concentrations, which can be as high as 0.4. These predictions will probably change as more enzymes are included in the model though. Also, by expanding the model, more accurate total enzyme predictions may be achieved, since, if free metabolite concentrations are lower because each metabolite participates in more than one reaction, then predicted total enzyme concentrations will likely be lower as well and closer to experimental values. However, eMASS2 still needs to be validated against metabolite time-course data. One reason we did not do such validation at this point, is because we consider the model to be too incomplete to be validated, with key glycolytic enzymes missing, e.g. PFK and PYK. Instead, we focused on comparing eMASS and eMASS2

Future work includes adding the remaining enzymes in *E. coli* glycolysis to the model and properly validating it against experimental data. Also the method used to combine the enzyme-level models into system-level models in eMASS2 should be improved, such that more enzyme-level models can be considered instead of just the best three. One option would be to just randomly sample combinations of enzyme-level models instead, another option would be to do the same in a more intelligent way with an evolutionary algorithm or similar. eMASS2 could, in principle, also be used to fit the rate constant values when no kinetic data is available for a given enzyme. Finally, an alternative to predicting free metabolite concentrations through eMASS2, which would considerably simplify the workflow, would be to somehow estimate the free metabolite concentrations at the genome-scale level for each metabolite in the model. These concentrations would then just be integrated into the models, avoiding the need to predict free metabolite concentrations.

.

3.5 Supplementary information

3.5.1 Metabolomics data

Metabolite	Concentration (M)
13DPG	1.65×10^{-5}
2PG	9.18×10^{-5}
23DPG	8.29×10^{-5}
3PG	1.54×10^{-3}
ADP	5.55×10^{-4}
ATP	9.63×10^{-3}
DHAP	3.06×10^{-3}
FDP	1.52×10^{-2}
G3P	2.71×10^{-4}
NAD	2.55×10^{-3}
NADH	8.36×10^{-5}
PEP	1.84×10^{-4}
PI	2.39×10^{-2}

3.5.2 Proteomics data

Isoenzyme	Reaction	Concentration (M)
fbaA	FBA2	5.22×10^{-6}
fbaB	FBA1	1.25×10^{-7}
gpmI	PGMi	1.27×10^{-6}
gpmA	PGMd	3.64×10^{-6}
eno	ENO	9.51×10^{-6}
gapA	GAPDH	6.02×10^{-6}
pgk	PGK	1.95×10^{-5}
tpi	TPI	1.89×10^{-6}

References

[1] A. Bar-Even, R. Milo, E. Noor, and D. S. Tawfik. The moderately efficient enzyme: futile encounters and enzyme floppiness. eng. *Biochemistry*, 54(32):4969–4977, 2015. ISSN: 15204995, 00062960. DOI: 10.1021/acs.biochem.5b00621.

[2] B. D. Bennett, E. H. Kimball, M. Gao, R. Osterhout, S. J. Van Dien, and J. D. Rabinowitz. Absolute metabolite concentrations and implied enzyme active site occupancy in *Escherichia coli*. *Nature chemical biology*, 5(8):593–9, Aug. 2009.

[3] B. D. Bennett, J. Yuan, E. H. Kimball, and J. D. Rabinowitz. Absolute quantitation of intracellular metabolite concentrations by an isotope ratio-based approach. *Nature protocols*, 3(8):1299–1311, 2008.

[4] E. Brunk, N. Mih, J. Monk, Z. Zhang, E. J. O'Brien, S. E. Bliven, K. Chen, R. L. Chang, P. E. Bourne, and B. O. Palsson. Systems biology of the structural proteome. *Bmc systems biology*, 10(1):26, 2016.

[5] A. B. Canelas, W. M. Van Gulik, and J. J. Heijnen. Determination of the cytosolic free NAD/NADH ratio in *Saccharomyces cerevisiae* under steady-state and highly dynamic conditions. *Biotechnology and bioengineering*, 100(4):734–743, 2008.

[6] S. Chandrasekaran and N. D. Price. Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in *Escherichia coli* and *Mycobacterium tuberculosis*. *Proceedings of the national academy of sciences*, 107(41):17845–17850, 2010.

[7] C. Chassagnole, N. Noisommit-Rizzi, J. W. Schmid, K. Mauch, and M. Reuss. Dynamic modeling of the central carbon metabolism of *Escherichia coli*. *Biotechnology and bioengineering*, 79(1):53–73, 2002.

[8] A. Chowdhury, A. R. Zomorrodi, and C. D. Maranas. k-OptForce: integrating kinetics with flux balance analysis for strain design. *Plos computational biology*, 10(2):e1003487, Feb. 2014.

[9] K. v. Eunen, J. A. L. Kiewiet, H. V. Westerhoff, and B. M. Bakker. Testing Biochemistry Revisited: How *In Vivo* Metabolism Can Be Understood from *In Vitro* Enzyme Kinetics. *Plos computational biology*, Jan. 2012.

[10] A. M. Feist, C. S. Henry, J. L. Reed, M. Krummenacker, A. R. Joyce, P. D. Karp, L. J. Broadbelt, V. Hatzimanikatis, and B. Ø. Palsson. A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular systems biology*, 3(121):1–18, 2007.

[11] D. Gutnick, J. M. Calvo, B. N. Ames, and T. Klopotowski. Compounds Which Serve as the Sole Source of Carbon or Nitrogen for *Salmonella typhimurium* LT-2 Compounds Which Serve as the Sole Source of Carbon or Nitrogen for *Salmonella typhimurium* LT-2. *Journal of bacteriology*, 100(1):215–219, 1969.

[12] Hnin W Aung, Susan A Henry and L. P. Walker. Revising the Representation of Fatty Acid, Glycerolipid, and Glycerophospholipid Metabolism in the Consensus Model of Yeast Metabolism. *Industrial biotechnology*, 9(4):215–228, 2013.

[13] N. Ishii, Y. Suga, A. Hagiya, H. Watanabe, H. Mori, M. Yoshino, and M. Tomita. Dynamic simulation of an *in vitro* multi-enzyme system. *Febs letters*, 581(3):413–420, 2007.

[14] N. Jahan, K. Maeda, Y. Matsuoka, Y. Sugimoto, and H. Kurata. Development of an accurate kinetic model for the central carbon metabolism of *Escherichia coli*. *Microbial cell factories*, 15(1):112, 2016.

[15] N. Jamshidi and B. Ø. Palsson. Mass action stoichiometric simulation models: incorporating kinetics and regulation into stoichiometric models. *Biophysical journal*, 98(2):175–85, Jan. 2010.

[16] D. Kesten, U. Kummer, S. Sahle, and K. Hübner. A new model for the aerobic metabolism of yeast allows the detailed analysis of the metabolic regulation during glucose pulse. *Biophysical chemistry*, 206:40–57, 2015.

[17] A. Khodayari and C. D. Maranas. A genome-scale escherichia coli kinetic metabolic model satisfying flux data for multiple mutant strains. *Nature communications*, 7:1–12, 2016. ISSN: 2041-1723. DOI: 10.1038/ncomms13806. URL: http://dx.doi.org/10.1038/ncomms13806.

[18] D. McCloskey, B. Ø. Palsson, and A. M. Feist. Basic and applied uses of genome-scale metabolic network reconstructions of *Escherichia coli*. *Molecular systems biology*, 9(1):661, 2013.

[19] P. Millard, K. Smallbone, and P. Mendes. Metabolic regulation is sufficient for global and robust coordination of glucose uptake, catabolism, energy production and growth in *Escherichia coli*. *Plos computational biology*, 13(2):e1005396, 2017.

[20] L. Miskovic and V. Hatzimanikatis. Production of biofuels and biochemicals: In need of an ORACLE. *Trends in biotechnology*, 28(8):391–397, 2010.

[21] M. L. Mo, B. Ø. Palsson, and M. J. Herrgård. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. *Bmc systems biology*, 3(1):37, 2009.

[22] M. Newville, T. Stensitzki, D. B. Allen, and A. Ingargiola. LMFIT: Non-Linear Least-Square Minimization and Curve-Fitting for Python. Sept. 2014. DOI: 10.5281/zenodo.11813. URL: https://doi.org/10.5281/zenodo.11813.

[23] E. J. O'Brien, J. a. Lerman, R. L. Chang, D. R. Hyduke, and B. Ø. Palsson. Genome-scale models of metabolism and gene expression extend and refine growth phenotype prediction. *Molecular systems biology*, 9(693):693, 2013.

[24] J. D. Orth, T. M. Conrad, J. Na, J. A. Lerman, H. Nam, A. M. Feist, and B. O. Palsson. A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism–2011. *Molecular systems biology*, 7(1):535–535, 2014.

[25] J. O. Park, S. A. Rubin, Y.-f. Xu, D. Amador-noguez, J. Fan, T. Shlomi, and J. D. Rabinowitz. Metabolite concentrations, fluxes and free energies imply efficient enzyme usage. *Nature chemical biology*, (May), 2016.

[26] K. Peskov, E. Mogilevskaya, and O. Demin. Kinetic modelling of central carbon metabolism in *Escherichia coli*. *Febs journal*, 279(18):3374–3385, 2012.

[27] P. A. Saa and L. K. Nielsen. Construction of feasible and accurate kinetic models of metabolism: A Bayesian approach. *Scientific reports*, 6(July):29635, 2016.

[28] P. Saa and L. K. Nielsen. A general framework for thermodynamically consistent parameterization and efficient sampling of enzymatic reactions. *Plos computational biology*, 11(4):e1004195, 2015.

[29] B. J. Sánchez, C. Zhang, A. Nilsson, P.-J. Lahtvee, E. J. Kerkhoven, and J. Nielsen. Improving the phenotype predictions of a yeast genome-scale metabolic model by incorporating enzymatic constraints. *Molecular systems biology*, 13(8):935, 2017.

[30] A. Schmidt, K. Kochanowski, S. Vedelaar, E. Ahrne, B. Volkmer, L. Callipo, K. Knoops, M. Bauer, R. Aebersold, and M. Heinemann. The quantitative and condition-dependent *Escherichia coli* proteome. *Nature biotechnology*, in press(December), 2015.

[31] K. Smallbone, H. L. Messiha, K. M. Carroll, C. L. Winder, N. Malys, W. B. Dunn, E. Murabito, N. Swainston, J. O. Dada, F. Khan, P. Pir, E. Simeonidis, I. Spasić, J. Wishart, D. Weichart, N. W. Hayes, D. Jameson, D. S. Broomhead, S. G. Oliver, S. J. Gaskell, J. E. G. McCarthy, N. W. Paton, H. V. Westerhoff, D. B. Kell, and P. Mendes. A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *Febs letters*, 587(17):2832–41, Sept. 2013.

[32] L. M. Tran, M. L. Rizk, and J. C. Liao. Ensemble modeling of metabolic networks. *Biophysical journal*, 95(12):5606–17, Dec. 2008.

[33] B. Volkmer and M. Heinemann. Condition-Dependent cell volume and concentration of *Escherichia coli* to facilitate data conversion for systems biology modeling. *Plos one*, 6(7):1–6, 2011.

should be

Chapter 4

Using molecular dynamics simulations to parameterize models of enzyme reaction kinetics

Abstract

Kinetic models of metabolism are central to systems biology and understanding the inner workings of cellular metabolism. Yet, models accounting for enzyme regulation are hard to parameterize due to a general lack of kinetic data. Even where available, multiple measured data may be inconsistent with general enzyme kinetic parameters or existing data on a given enzyme. To overcome this challenge, we introduce a novel method which pairs molecular dynamics simulations, which estimate free energy of binding differences using molecular mechanics, and enzyme-level kinetic models for *E. coli*. Here, we use MM-PBSA to estimate the relative binding energy, $\Delta\Delta G$, between substrate(s) and product(s) for three enzymes in *E. coli*'s central carbon metabolism, and integrate this calculated $\Delta\Delta G$ with the respective enzyme-level metabolic kinetic models. We perform this analysis using the elementary Mass Action Stoichiometric Simulation (eMASS) modeling framework, which allows for integration of different kinds of kinetic data as constraints on the model. Using this platform, we assess the consistency of the computed $\Delta\Delta G$ with other enzyme-specific kinetic data and the degree to which this $\Delta\Delta G$ constrains the sampled elementary rate constants of the enzyme kinetic models. We compare the impact of this constraint to that of integrating other kinetic parameters, such as a k_{cat} or a K_m. We find that the calculated $\Delta\Delta G$ values are compatible with the remaining enzyme-specific kinetic data used here. Also, it is generally as constraining as other kinetic/thermodynamic parameters, e.g. K_m, k_{cat}, or K_{eq}. Thus MM-PBSA calculated parameters provide a promising alternative to parameterize reactions with missing data. Code and results are available at https://github.com/martamatos/eMASS-MD.

4.1 Introduction

Calculating binding energies of small molecules to proteins for drug design is a typical application of molecular dynamics methods [72, 69, 43, 9]. However, a change in binding affinity can have consequences at the cellular level as well, which these methods cannot capture. To capture such effects, one needs to integrate the results of atomistic methods with methods that can model the whole network around the protein of interest [62], e.g. signaling or metabolic networks. The potential for integration of atom-level simulations and cellular systems biology has been demonstrated by Gabdoulline *et al* [14], who used Brownian dynamics simulations to estimate an undetermined rate constant for the association of superoxide with peroxidase in the peroxidase-oxidase reaction. Also, more recently, Mih and Brunk [40] integrated molecular dynamics methods with a metabolic model of the human red blood cell. They first quantified the impact of enzyme mutations on drug binding affinities, and then integrated the difference in drug binding affinities with a metabolic model of the human red blood cell to understand the impact in the cellular metabolism. Such a multi-scale approach can be very useful in the age of personalized medicine, e.g. to take into account an individual's mutations when prescribing a given drug.

Another field where assessing the impact of enzyme mutations on cell metabolism is important is metabolic engineering, where the goal is typically to increase the production of a given molecule in a given host organism. To do so, one may follow a rational strain design approach where metabolic models are used to decide on the strategy to follow [70, 23, 37], e.g. which genes to knockout or to insert, or which enzymes to engineer.

Two main types of metabolic models are: 1) stoichiometric models [52], usually built at genome-scale, these consist of the set of biochemical reactions encoded in the organism's genome and are assumed to be at steady-state, i.e. metabolite concentrations are constant and not state variables of the system, thus only steady-state fluxes can be predicted; 2) kinetic models [2, 8], typically on a smaller scale, these model the kinetics of the involved enzymes in detail, and therefore enable time-resolved predictions of metabolite concentrations and reaction fluxes.

One key advantage of kinetic models over stoichiometric models is their ability to model substrate-level enzyme regulation, e.g. allosteric regulation, competitive inhibition, or substrate activation, which ultimately depend on the concentration of the metabolites involved and is therefore outside the scope of stoichiometric models.

Kinetic models, on the other hand, are hard to build due to the amount of parameters involved. This is where integration with computational approaches that can estimate kinetic parameters from three-dimensional protein structures is important [63]. Here, relative binding energies estimated through molecular dynamics simulations can be used not only to assess the impact of enzyme mutations on metabolism, but to parameterize the model in the first place.

Different types of methods are available to calculate relative binding energies: docking and scoring methods, which are fast but not very accurate. Alchemical methods [60], e.g. FEP or TI, on the other hand are known to be precise but time-consuming, also presenting some limitations regarding ligand similarity and charge [17]. More recently, end-point methods have been introduced, such as LIE [22, 3], LRA, PDLD/S, PDLD/S-LRA [36, 38, 27, 58], or MM-PBSA/GBSA [61, 31]. Several studies have been performed to compare the accuracy and precision of different alchemical and end-point methods, giving mixed results, with different methods performing better in different systems [16, 4, 32, 24, 55, 43, 41, 34, 21, 66, 20, 47, 26, 18, 57, 15].

Here, we use MM-PBSA to calculate relative binding energies, as it offers a good trade-off between speed and accuracy/precision [17, 50]. We choose three enzymatic reactions in the *Escherichia coli* metabolism and calculate the binding energy of substrates and products to the respective enzymes. We then subtract the binding energy for the substrate(s) to the binding energy for the product(s), to obtain a relative binding energy, $\Delta\Delta G$. Each relative binding energy is then converted into a relative binding constant, ΔK_b, which is readily integrated with an enzyme-level kinetic model for the respective reaction. In this kinetic model reactions are decomposed into elementary steps, where each step is modeled by a mass action rate law. This leads to several elementary rate constants, which are fitted to macroscopic kinetic data. By integrating ΔK_b with the enzyme-level kinetic model, we expect to constrain the sampling space for the reaction's rate constants. Once ΔK_b is integrated in the respective enzyme model, we 1) assess if its compatibility with the kinetic data already used to fit the model, 2) assess how much it constrains the model's rate constants sampling space, and finally 3) how it impacts the dynamic behavior of the system.

4.2 Modeling *E. coli*'s enzyme kinetics with mass action reaction mechanisms

A common approach to model the dynamics of metabolism is to model each reaction in the system using Michaelis-Menten rate laws, whose parameters consist mostly of turnover rates, k_{cat}, and Michaelis-Menten constants, K_m, which are usually measured in *in vitro* experiments.

One drawback of this approach is that it relies on the quasi-steady-state assumption, according to which the concentration of the intermediate compounds in the reaction barely changes while the reaction takes place. To avoid this and other issues, we use the elementary Mass Action Stoichiometric Simulation (eMASS) framework introduced in chapter 2, to build a kinetic model for *E. coli*'s central carbon metabolism. The eMASS framework is a more general bottom-up approach, where each metabolic reaction is modeled independently, and then all enzyme-level models can be integrated to form a final model for the metabolic system of interest. This system-level model can then be further constrained by integrating omics data. Each individual enzyme-level model in particular is built by decomposing the reaction into elementary steps, according to its mechanism, i.e. the order in which substrates and products bind to and are released from the enzyme, respectively, which we either obtain from literature or assume to be random. Each elementary reaction step is modeled with mass action kinetics and no assumptions are made. Yet, this results in several rate constants whose values we do not know, thus, to estimate them we fit each enzyme model to macroscopic kinetic parameters, such as K_{eq}, k_{cat}, K_m, or K_i.

Two key advantages of the eMASS framework is that it does not assume any particular reaction mechanism and it can integrate kinetic parameters from different sources, e.g. two k_{cat} measurements with different cosubstrate concentrations. Furthermore, any parameter defined in terms of rate constants can be readily integrated in the model.

However, a common issue with building kinetic models following a bottom-up approach is the lack of macroscopic kinetic parameters. Thus, here we test if using MM-PBSA to estimate relative binding constants is a viable approach to overcome this lack of data.

To do so, we follow the workflow detailed in Figure 4.1: 1) once the relative binding energy between substrate(s) and product(s) has been calculated, we convert it into a relative binding constant, ΔK_b, by using the following relation between the Gibbs energy and binding constant:

$$\Delta G = -RT \ln (K_b)$$

where ΔG is in kcal mol^{-1}, R is the relative gas constant in kcal K^{-1} mol^{-1}, T is the absolute temperature in Kelvin, and K_b is a binding constant. This relative binding constant can be expressed in terms of rate constants and integrated in the enzyme-level kinetic model. See section 4.7.2 for details on the $\Delta\Delta G$ conversion to ΔK_b for each enzyme. 2) We perform a parameter scan over ΔK_b, to check which values integrate better with the enzyme's kinetic data. In particular, if the value estimated through MM-PBSA is in that range. 3) Add ΔK_b to the data used to fit the enzyme, generate 100 model ensembles with 100 models each. 4) Filter out models that do not fit properly, the criterion being defined by a cutoff on the sum of squared errors between the logarithm of the data used to fit the model and the logarithm of the predicted values. 5) Analyze the impact of ΔK_b both on each rate constant interquartile range and on the possible combinations of rate constants in each model ensemble. 6) Analyze the impact of ΔK_b on the reaction's dynamics. Finally, because simulations of enzyme models not constrained by a k_{cat} can take an irrealistically long time to converge ($\sim$ 3000 years), we add an estimated k_{cat} value for ENO, which was not part of the original dataset.

4.3 Molecular Dynamics simulations to parameterize *E. coli* enzyme-level kinetic models

We choose three enzymes in *E. coli*'s central carbon metabolism to test the integration of MD simulations with metabolic kinetic models. The enzymes are enolase (ENO), glyceraldehyde-3-phosphate dehydrogenase (GAPDH), and transaldolase B (TALB). ENO and GAPD are part of glycolysis, while TALB is part of the pentose-phosphate pathway. ENO has only one substrate, 2-phospho-D-glycerate (2PG), and one product, phosphoenolpyruvate (PEP), thus the binding energy is stimated for both metabolites and $\Delta\Delta G = \Delta G_{PEP} - \Delta G_{2PG}$. There is little kinetic data available for this enzyme: an equilibrium constant, K_{eq}, obtained from the eQuilibrator web-server [12, 44], and a Michaelis-Menten constant, K_m. GAPD has 3 substrates: nicotinamide adenine dinucleotide (NAD), glyceraldehyde 3-phosphate (G3P), and phosphate (PI), and 2 products: nicotinamide adenine dinucleotide reduced (NADH)

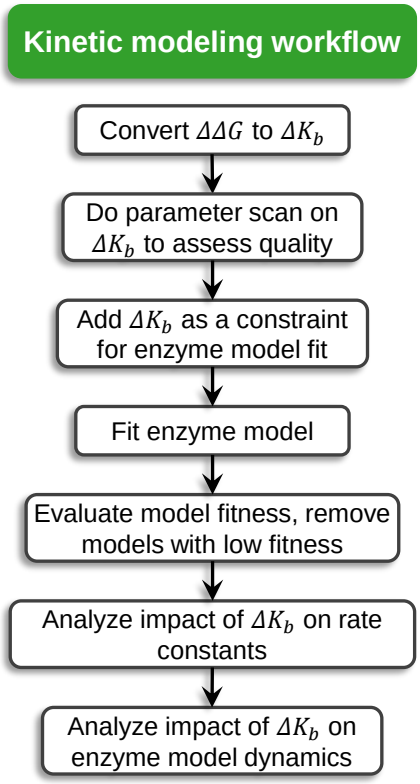


Figure 4.1: Workflow used to assess the impact of $\Delta\Delta G$ on a metabolic kinetic model at the enzyme level.

and 3-phospho-D-glyceroyl phosphate (13DPG). We calculate the binding energy for the complex NAD-G3P and then the binding energy for NADH-13DPG, thus the difference in binding energy is $\Delta\Delta G = \Delta G_{NAD-G3P} - \Delta G_{NADH-13DPG}$. Regarding the available kinetic data for GAPD, there is a K_m for NAD, G3P, and PI, a k_{cat} for the forward direction, a K_{eq}, and a dissociation constant for NAD. Finally, TALB has 2 substrates: D-sedoheptulose 7-phosphate (S7P) and D-glyceraldehyde 3-phosphate (G3P) and 2 products: D-fructose-6-phosphate (F6P), and D-erythrose 4-phosphate (E4P),. Because this enzyme follows a ping pong mechanism, we estimate the binding energy only for the first metabolite to bind and the last metabolite to be released, as we did not model the enzyme structural change upon binding of the first metabolite. Thus, $\Delta\Delta G = \Delta G_{S7P} - \Delta G_{F6P}$. For TALB, there is a K_{eq}, a K_m for each metabolite, two k_{cat} for the forward direction, and an inhibition constant for phosphate, which is competitive with regard to all metabolites. For the detailed enzyme mechanisms, see figure 4.3. With these three enzymes, we can assess what is the impact of including the $\Delta\Delta G$ in three different scenarios: 1) in enzyme models for which there is little kinetic data available but the $\Delta\Delta G$ includes all substrates and products, thus tightly

constraining the reaction's conversion step when combined with the reaction's K_{eq} (ENO); 2) when there is enough kinetic data available and we constrain the binding of almost all substrates vs all products (GAPD), which means ΔK_b includes most rate constants in the enzyme model; 3) when there is enough kinetic data available but we only constrain the binding of only the first substrate vs. the last product (TALB), which means ΔK_b includes less than half the rate constants in the enzyme model.

To calculate $\Delta\Delta G$ for all enzymes, we follow the general workflow in Figure 4.2, where we 1) first select a wild type enzyme crystal structure from the protein data bank [6] (see table 4.2 in the methods section for more details); 2) perform a MD simulation of the chosen crystal structure for $\sim$ 100 ns to obtain an ensemble of representative enzyme conformations. For ENO we simulate only the holo form with two Mg^{2+} ions bound, for GAPD we perform two simulations for the holo form, one with NAD bound and the other with NAD and G3P bound, and for TALB we simulate both the apo form and the holo form with S7P. This leads to 3) an ensemble of representative enzyme conformations with $\sim$ 700 enzyme conformations (where G3P and S7P were removed from the respective conformations), to which we 4) dock the substrate and product separately using DOCK6 [35]. For ENO we dock 2PG and PEP; for GAPD, we dock G3P and 13DPG; for TALB, we dock F6P and S7P. 5) We cluster the $\sim$ 700 docking poses according to the distances between three chosen atoms in the ligand and three chosen atoms in the binding residues, take between four and ten of the largest clusters, and pick representative frames from each one such that we have between 40 and 50 docking poses for each enzyme-ligand complex, which we 6) use as a starting point for an MD simulation of $\sim$ 1 ns. The first 0.1 ns is discarded from this MD simulation, as the complex is considered to be still equilibrating during this time, and the remaining 0.9 ns are used to 7) do MM-PBSA and obtain the binding energy for the enzyme-ligand complex. Finally, we 8) subtract the binding energy for the substrate from the product and obtain a $\Delta\Delta G$. The reason we are interested in the $\Delta\Delta G$ instead of the binding energy for a given substrate or product, is because this way we can avoid calculating the entropy term in step 7 [39, 17]. Calculating the entropy term in step 7 would lead to longer running times for the MM-PBSA algorithm [42, 25] and probably to higher uncertainties as well [25, 17].

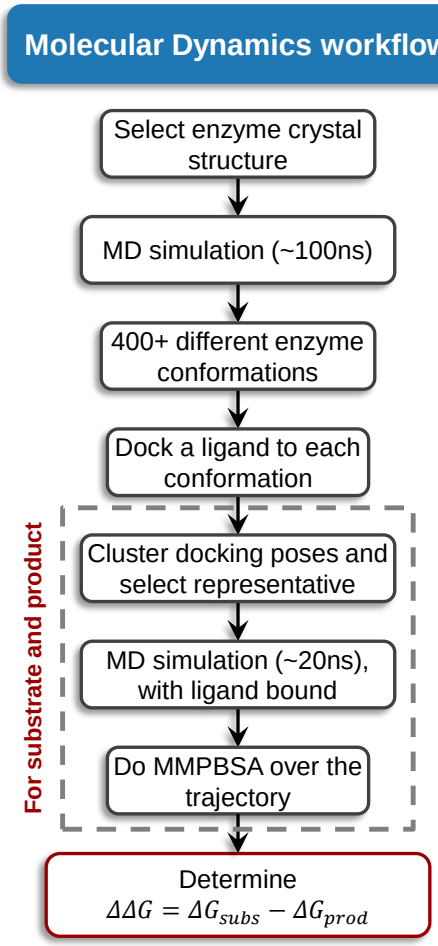


Figure 4.2: Workflow used to estimate $\Delta\Delta G$ values for substrate(s) vs product(s) of 3 enzymes using molecular dynamics methods.

4.4 Results

4.4.1 $\Delta\Delta G$ calculation using MM-PBSA

In table 4.1, we show the $\Delta\Delta G$ and respective standard error and ΔK_b values obtained for ENO ($\Delta G_{PEP} - \Delta G_{2PG}$), GAPD ($\Delta G_{NAD-G3P} - \Delta G_{NADH-DPG}$), and TALB ($\Delta G_{F6P} - \Delta G_{S7P}$), by following the workflow in Figure 4.2. The standard error for each $\Delta\Delta G$ value is propagated from the respective ΔG values, which in turn is taken as the standard error of the mean ΔG across the respective MD simulations. The ΔG values obtained from each MD simulation for each enzyme-ligand complex

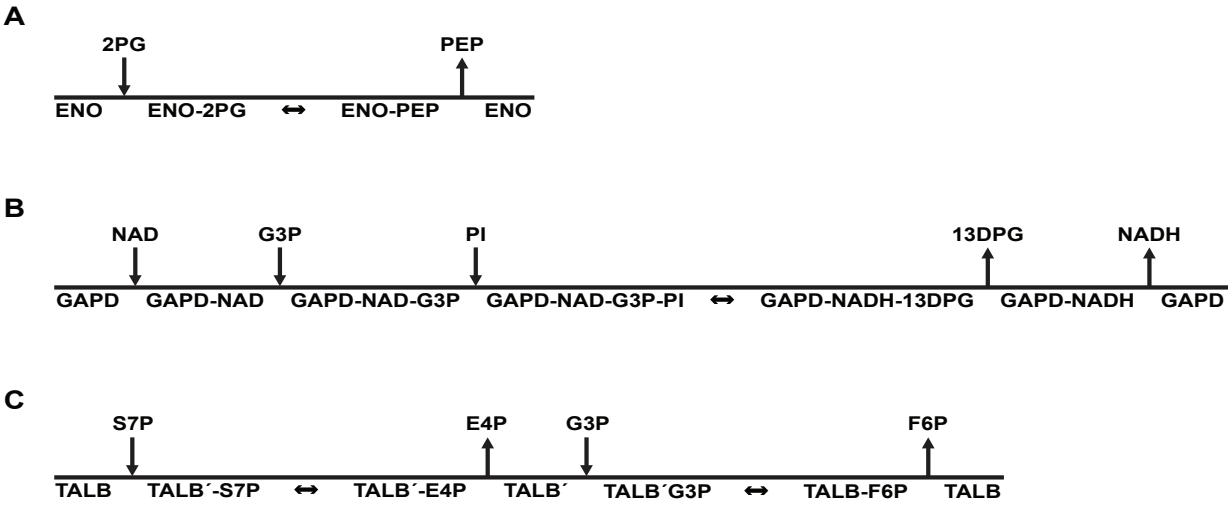


Figure 4.3: Enzymatic reaction and respective mechanisms for substrate binding and product release for (A) ENO, (B) GAPD, and (C) TALB. Enzyme-ligand complexes are denoted $enzyme - ligand$.

Enzyme	$\Delta\Delta G_{mean}(kcal/mol)$	$\Delta\Delta G_{sem}(kcal/mol)$	ΔK_b
ENO	-11.58	1.84	3.6×10^{-9}
GAPD	-0.41	1.77	0.5
TALB	8.39	1.92	1.3×10^{6}

Table 4.1: $\Delta\Delta G$ values and respective standard error of the mean and ΔK_b for each enzyme.

are shown in section 4.7.1. For GAPD and TALB we estimate ΔG for each enzyme-ligand complex from 40 to 50 independent MD simulations (apo and holo enzyme forms, four to five clusters per enzyme form, five to ten representative frames per cluster). For ENO, we first used the crystal structure 1E9I [33], subunit D with two Mg^{2+} ions bound and followed the same workflow, at first with one enzyme form, three clusters per enzyme form, and four representative frames per cluster. However, the variance among ΔG values is very high, see Fig. 4.4. This might be due to the presence of the metal ions and their influence on the docking of 2PG and PEP [72]. Besides, the second Mg^{2+} ion is known to bind only after the ligand [33]. Hence, a better approach would have been to bind the second Mg^{2+} ion manually after docking 2PG and PEP with DOCK6 [35]. Therefore, we use the crystal structure 3H8A [45] instead, and bind the second Mg^{2+} ion manually as well as 2PG and PEP, run a ≈ 75

ns MD simulation, take one frame every $\approx$ 1 ns from the last 25 ns of simulation, cluster all 250 frames, take the five largest clusters, and pick 10 to 20 representative from each cluster to start new MD simulations, which are used to calculate the Gibbs binding energies.

For all enzymes we discarded all positive ΔG values, as we do not think these represent productive binding events, in the sense that a binding conformation with $\Delta G < 0$ would probably not lead to the conversion of the complex enzyme-substrate to the complex enzyme-product. Also, for the binding of PEP to ENO, the ΔG_{PEP} distribution seems to be bi-modal, see Fig. 4.5, with a group of mean values of ~ 30 kcal mol^{-1}, and a second group of ~ 100 kcal mol^{-1}. Given the results for the binding of 2PG, we consider the second group of binding energies for PEP to not be realistic and remove it from the final ΔG_{PEP} calculation. Therefore, for ENO, the final $\Delta\Delta G$ is calculated based on 50 simulations for the complex ENO-2PG and 43 simulations for ENO-PEP. For GAPD we calculate the final $\Delta\Delta G$ based on 50 MD simulations for each enzyme-ligand complex, and for TALB the final $\Delta\Delta G$ is calculated based on 45 MD simulations for the complex TALB-S7P and 40 simulations for the complex TALB-F6P. The starting frames for each enzyme-ligand complex can be obtained at: https://github.com/martamatos/eMASS-MD.

4.4.2 Integrating $\Delta\Delta G$ with enzyme-level kinetic models

ΔK_b impact on model fitness

To check if the ΔK_b value calculated with MM-PBSA is compatible with the other macroscopic kinetic parameters, we follow the workflow in figure 4.1 and start by doing a parameter scan over ΔK_b from 10^{-12} to 10^{12}. The quality of the fit of each model is measured using the sum of squared log-deviations defined as:

$$ssld = \sum_{i=0}^{n} (log(x_{data_i}) - log(x_{predicted_i}))^2$$

The reason to take the logarithm of the data to be fitted and the predicted data is that, for very small data values, e.g. 10^{-10}, even if the deviation from the data is one or two orders of magnitude, the typical sum of squared deviations is still small. Thus, wrongly giving the impression that the models fit the data well.

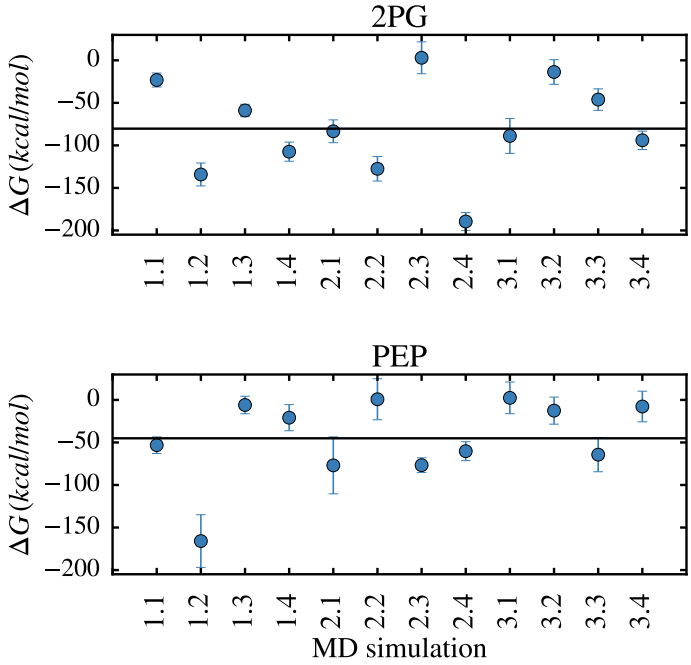


Figure 4.4: ΔG values for each ENO-ligand complex MD simulation when using the 1E9I structure. The starting frames for these simulations where selected from three different clusters, three frames per cluster, where e.g. 1.2 means cluster 1, frame 2. The error bars represent the standard deviation associated with the mean value.

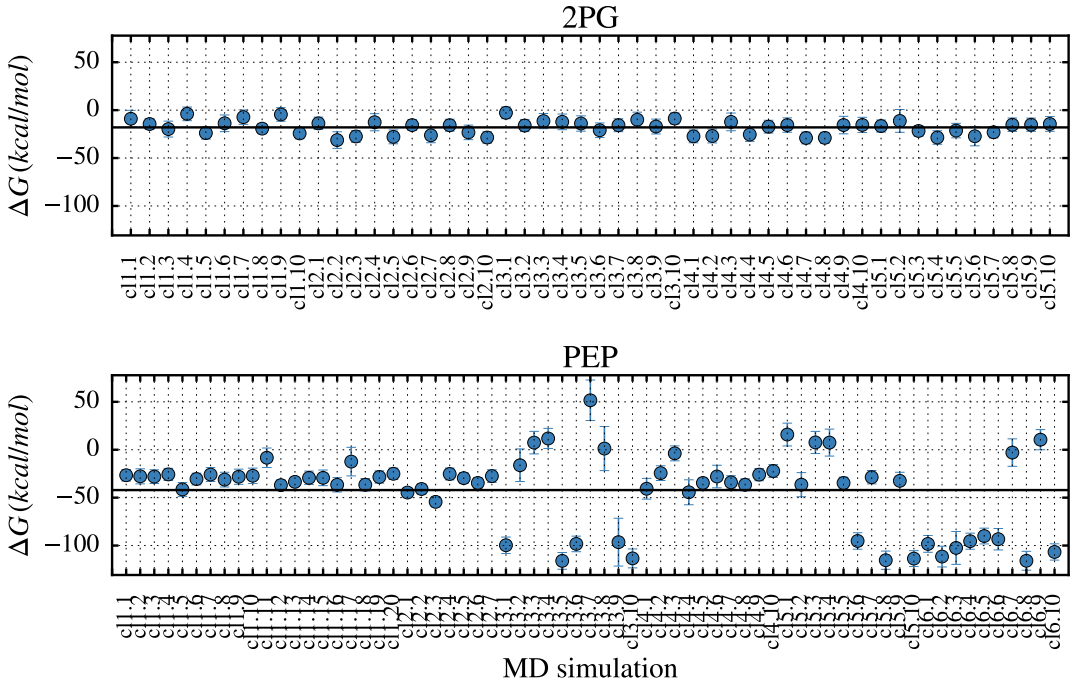


Figure 4.5: ΔG values for each ENO-ligand complex MD simulation when using the 3H8A structure. The starting frames for these simulations where selected from five different clusters for 2PG and six for PEP, five to ten frames per cluster, where e.g. 1.2 means cluster 1, frame 2. The error bars represent the standard deviation associated with the mean value.

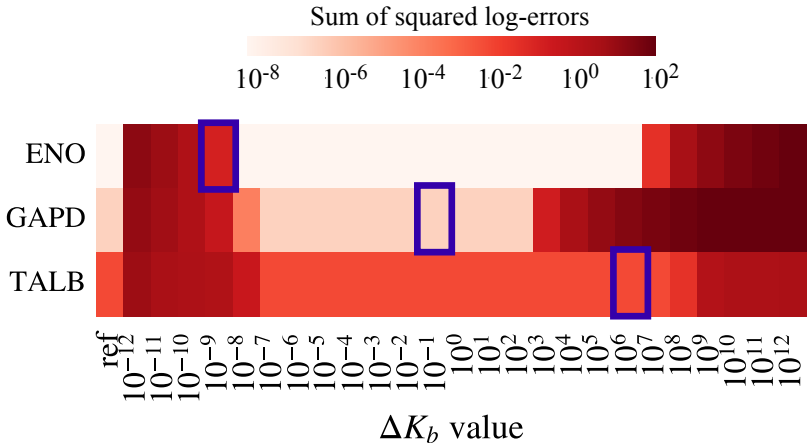


Figure 4.6: Parameter scan results over ΔK_b for ENO, GAPD, and TALB, which are represented in the rows, while the columns correspond to the different values of ΔK_b used. The first column corresponds to the $ssld_{median}$ when ΔK_b is not part of the fitting dataset. White corresponds to low $ssld_{median}$ values, i.e. higher fitness, while dark red corresponds to high ssd_{median} values, i.e. lower fitness. The value closest to the true ΔK_b for each enzyme is surrounded by a blue square.

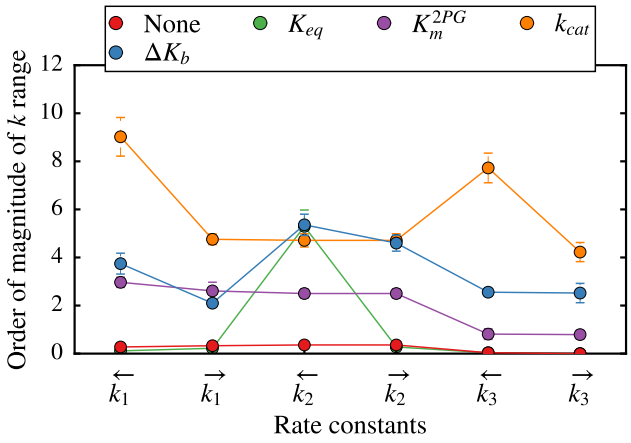
To measure the impact of each ΔK_b value on the fit quality, for each ΔK_b value, we generate 10 model ensembles, each with a 100 models. We then take the best ssld value for each model, and the median over these for each ensemble. Finally, we take the median of the resulting 10 median values (one per model ensemble), which corresponds to the $ssld_{median}$ values plotted in Figure 4.6. In the first column of Figure 4.6, we show the $ssld_{median}$ when ΔK_b is not included in the dataset used to fit the enzyme. This reference value is the same for GAPD and TALB than when ΔK_b is included in the fitting dataset, but seems to be lower for ENO. Regarding the effect of changing ΔK_b's value, generally there is a wide range of values that are compatible with the rest of the fitting dataset. The ΔK_b for ENO, 3.6×10^{-9}, is actually at the point where lower values are increasingly incompatible with the rest of the fitting dataset. However, when it is included in the dataset the ssld is still on the order of 5×10^{-9}, so it is still compatible with the remaining data points used for the fitting. This can also be seen in section 4.7.4, where we show all *ssld* values for all models when a given parameter is removed from the fitting dataset. In particular in Fig. 4.19a, we can see that the ssld when ΔK_b is removed from the fitting dataset does not change considerably. Hence, we consider the estimated ΔK_b values through MM-PBSA to be valid in the context of enzyme-level metabolic kinetic models.

ΔK_b impact on rate constant values

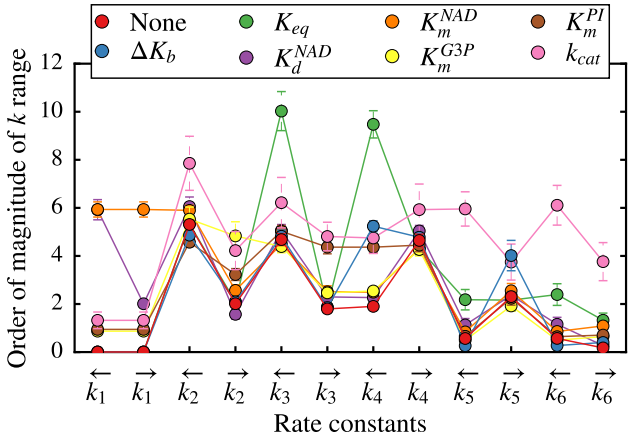
After evaluating the validity of the ΔK_b values in the context of metabolic kinetic models, we now check whether the rate constants sampling space was constrained or not when including ΔK_b in the enzyme fitting dataset. To do so, we generate 100 model ensembles, each with a 100 models, and discard all models with an ssld lower than 0.1[1]. Models with ssld lower than 0.1 will be referred to as valid models from here on. Also, only models with ssld lower than 0.1 are used in further analyses. For the valid models, we calculate the interquartile range (IQR) for each rate constant in each model ensemble and take the median IQR value over all model ensembles plus respective median absolute deviation, and check how these vary when different data points are removed from the fitting dataset. The results are presented in Figure 4.7. Generally, the parameters with greater impact on all rate constants IQR are the the k_{cat} and the K_{eq}. Yet, for ENO ΔK_b has a greater impact than K_{eq}, for GAPDo has less impact, and for TALB it does not considerably constrain the IQR of the rate constants, except for the first reverse and the last forward rate constants, which are part of the rate constant ratio constrained by the ΔK_b.

Besides analyzing how much ΔK_b restricts the rate constants IQR, we also ask if the number of possible combinations of rate constants in a given model ensemble is affected by ΔK_b. To answer this question, we start by doing a hierarchical clustering of all the valid models in a given model ensemble by rate constant values, and the difference is remarkable in the case of ENO. In Figure 4.8 we show an example of such clusterings for ENO where (a) ΔK_b was not part of the enzyme fitting dataset, and (b) ΔK_b was part of the enzyme fitting dataset. Basically, once the ΔK_b is included in the enzyme fitting dataset, there are only a few rate constant combinations that can fit the kinetic/thermodynamic data, as opposed to several rate constant combinations when ΔK_b is not part of the enzyme fitting dataset. The differences for GAPD and TALB are not as clear (see section 4.7.3), most likely because: 1) the proportion of rate constants constrained by ΔK_b is smaller than for ENO, and 2) these are constrained by more data points than ENO. Moreover, one reason for the combination of valid rate constant sets to decrease so dramatically in ENO is that the ΔK_b value is very close to the threshold where lower values become incompatible with the remaining fitting

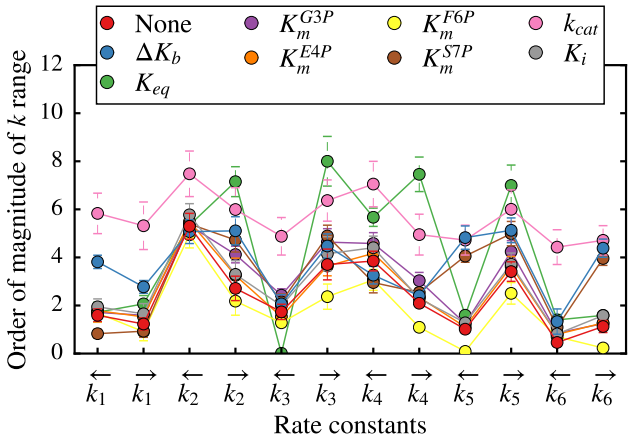
[1]By setting the ssld threshold at 0.1, all models reproduce the kinetic/thermodynamic data for the fitting with an error up to 10% and some outliers below 40%, see https://github.com/martamatos/eMASS-MD for details.



(a) ENO

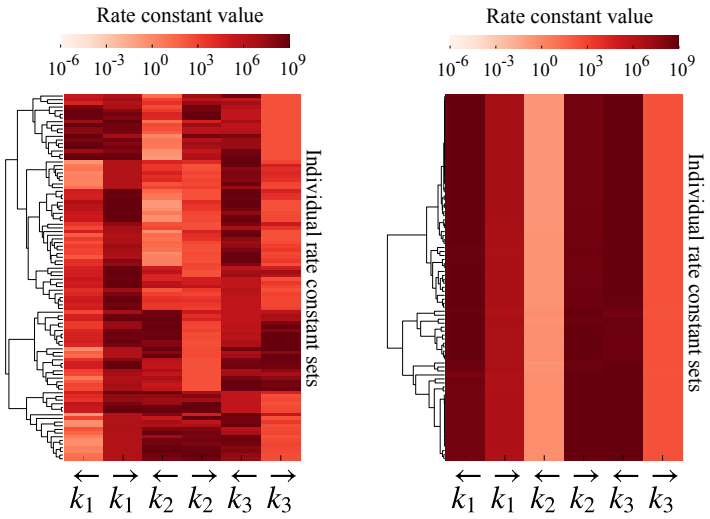


(b) GAPD



(c) TALB

Figure 4.7: Order of magnitude of the rate constants' median interquartile range (IQR) over each enzyme model ensemble when different kinetic parameters are removed from the fitting dataset one at a time. Red dots/lines represent values obtained when all parameters were included in the fitting dataset, blue dots/lines represent values obtained when ΔK_b was removed from the fitting dataset, and so on. The errorbars represent the median absolute deviation associated to the respective median IQR value.



(a) Without ΔK_b (b) With ΔK_b

Figure 4.8: Clustermap over all rate constant sets for one model ensemble for ENO: (a) when ΔK_b is not included in the fitting dataset and (b) when ΔK_b is included . Each row represents one rate constant set in the model ensemble, and each column represents one rate constant. The color represents the rate constant's order of magnitude: white represents a lower order of magnitude ($\approx 10^{-6}$) while darker red represents higher order of magnitude (up to 10^9).

dataset, since ΔK_b values on the order of 10^{-5} for ENO are constraining but not as much. See https://github.com/martamatos/eMASS-MD for all clustering results.

To quantify how much ΔK_b restricts the number of possible combinations of rate constants, k, in a given model ensemble, we discretize the rate constant values by binning them into the following bins: $[10^{-6}, 10^{-3}[$, $[10^{-3}, 10^{0}[$, $[10^{0}, 10^{3}[$,$[10^{3}, 10^{6}[$, $[10^{6}, 10^{9}[$, since $k \in [10^{-6}, 10^{9}]$. The combination of the bins that form one model is a pattern. We use Shannon's entropy [59] to quantify how many different patterns, i.e. how many different combinations of discretized rate constant values exist in one model ensemble. Shannon's entropy is defined as:

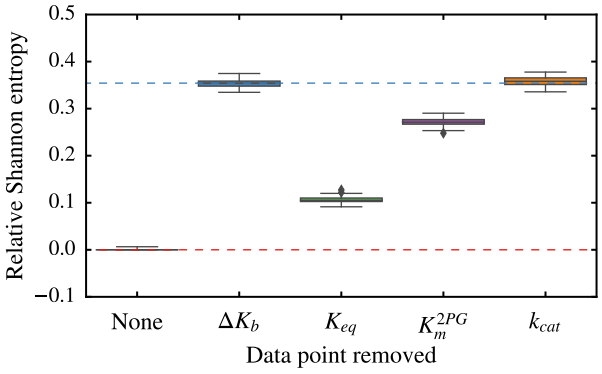
$$H(X) = \sum_{i=1}^{n} P(x_i) \log_{10} P(x_i)$$

where x_i is one pattern and $P(x_i)$ the probability of observing that pattern in a model ensemble with n models. To have a more meaningful metric, we use the relative Shannon entropy, which is calculated by dividing $H(x)$ by the maximum

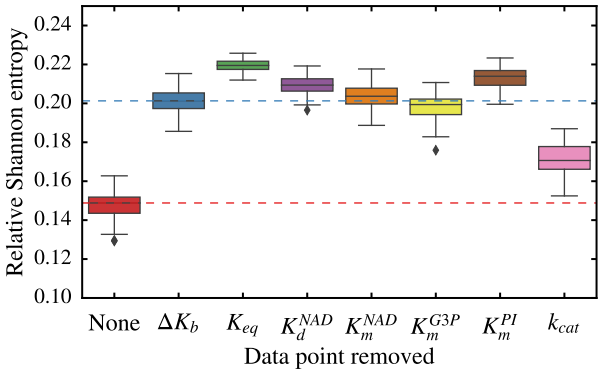
possible entropy, which corresponds to observing all the possible patterns: $\log_{10} 5^n$, where n is the number of rate constants in a model and 5 is the number of bins.

We compare how much the relative Shannon entropy is decreased by including ΔK_b in the dataset used to fit the enzyme to how much it decreases by including any of the other parameters. We do this comparison by answering the following question: having a complete dataset to fit the enzyme parameters, how much does the Shannon entropy increase when one data point is removed from the complete dataset? To answer this question, we take the complete dataset, remove one data point at a time, generate a 100 model ensembles for each of those incomplete datasets (where one data point was removed), and calculate the Shannon entropy for each of the 100 model ensembles generated from each incomplete dataset. This results in a 100 Shannon entropy values for each incomplete dataset. Also here, for each model ensemble, we only consider models with ssld < 0.1.

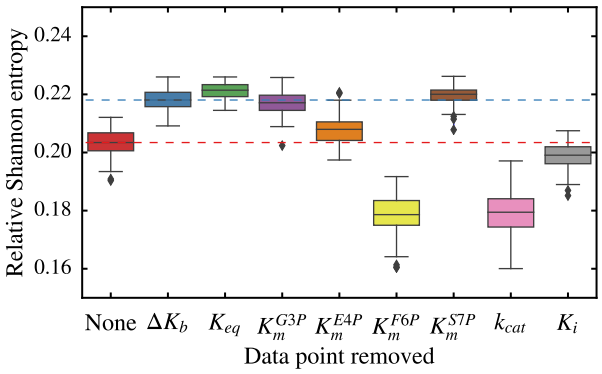
The Shannon entropy results for the three enzymes are shown in Figure 4.9. From these we can see that ΔK_b can considerably reduce the Shannon entropy in the rate constant combinations. Overall, ΔK_b can be as constraining as a k_{cat} or a K_m. ENO seems to be an extreme case, where including ΔK_b in the enzyme fitting dataset leads to only a few possible rate constant patterns. For TALB removing either the k_{cat}, K_i, or K_m from the fitting dataset reduces the Shannon entropy rather than increase it. By looking at the number of valid models per ensemble in section 4.7.5, we see that removing the k_{cat} or the K_i from TALB's fitting dataset, reduces the number of valid models. The same thing happens for ENO when the K_{eq} is removed from the fitting dataset and for GAPD when k_{cat} is removed from the fitting dataset. Thus, we calculate the relative Shannon entropy again considering only 50 models per ensemble for ENO, and 30 models per ensemble for GAPD and TALB. This way any change of entropy for each model ensemble shouldn't be related to the actual ensemble size. The results are shown in Figure 4.10. Overall the effect of including the ΔK_b in the enzymes fitting datasets is still comparable to that of any other parameter. For TALB removing the K_m for F6P still reduces the entropy, perhaps because removing it changes the fitness landscape in such a way that the amount of equivalent optima is reduced. We do recognize, however, that more models should be generated for each model ensemble, such that each ensemble contains at least 100 models with ssld < 0.1.



(a) ENO

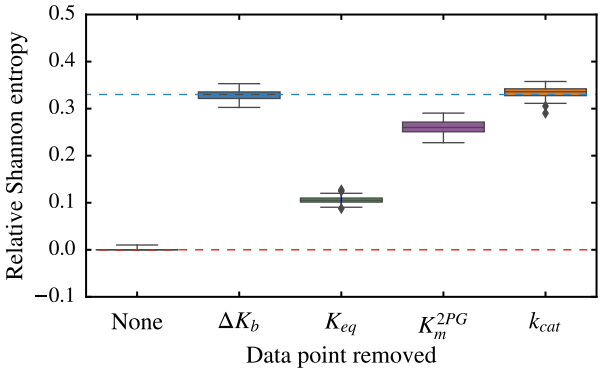


(b) GAPD

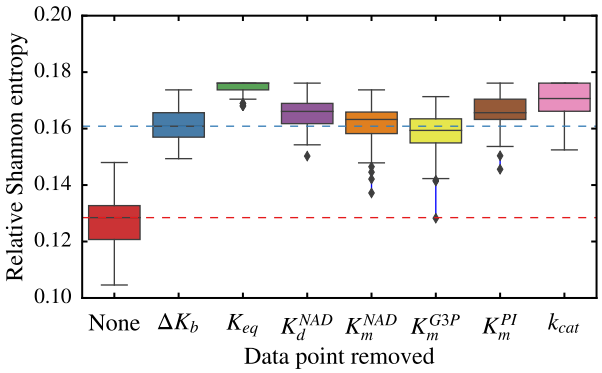


(c) TALB

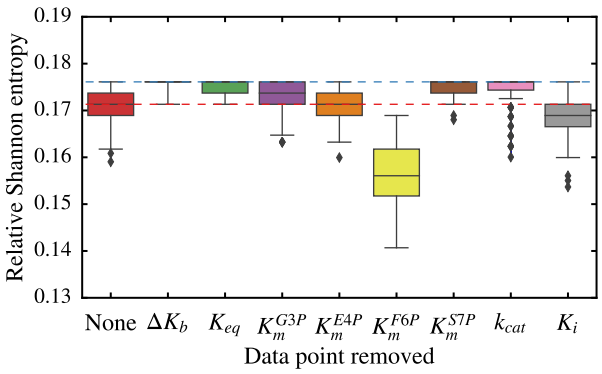
Figure 4.9: Relative Shannon entropy results over 100 model ensembles for each enzyme. Each box represents the relative Shannon entropy when a given data point is removed from the fitting dataset. In red all data points were used to fit the enzyme, in blue ΔK_b was removed from the fitting dataset, and so on. Each model ensemble contains up to 100 models with ssld < 0.1. The red line represents the median Shannon entropy value when the complete dataset is used to fit the enzyme. The blue line represents the median Shannon entropy value when ΔK_b is removed from the fitting dataset.



(a) ENO



(b) GAPD

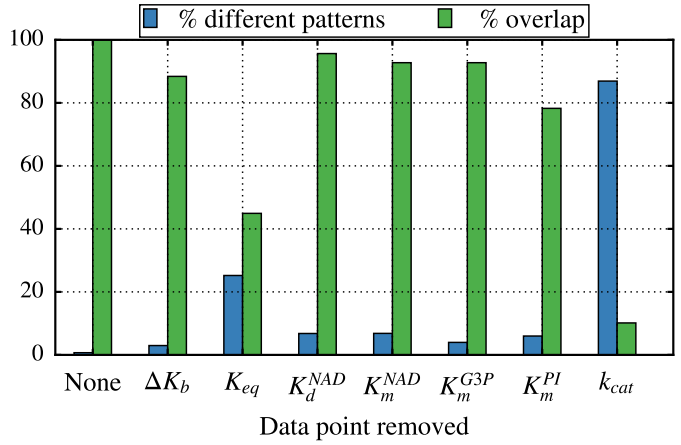


(c) TALB

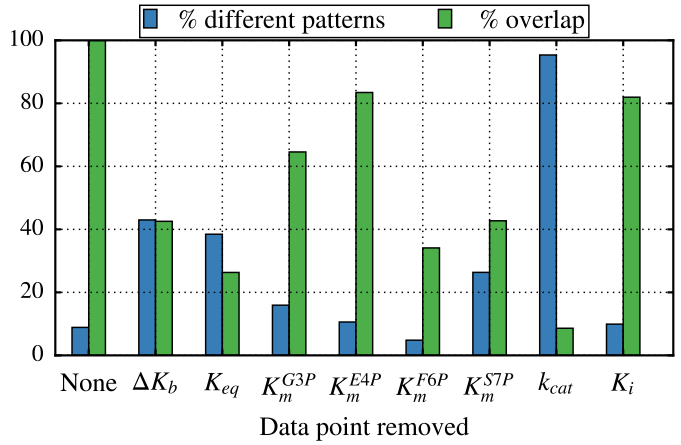
Figure 4.10: Relative Shannon entropy results over 100 model ensembles for each enzyme, however, here each model ensemble contains only up to 30 models with ssld < 0.1 for GAPD and TALB, and up to 50 models for ENO. Each box represents the relative Shannon entropy when a given data point is removed from the fitting dataset. In red all data points were used to fit the enzyme, in blue ΔK_b was removed from the fitting dataset, and so on. The red line represents the median Shannon entropy value when the complete dataset is used to fit the enzyme. The blue line represents the median Shannon entropy value when ΔK_b is removed from the fitting dataset.

The results from chapter 2 suggest however that, for enzymes with a high number of rate constants such as GAPD and TALB, the rate constant sets obtained in each model ensemble might not be reproducible. Hence, for each enzyme and each fitting dataset, we took all the model ensembles and checked how many different rate constant patterns were found as a percentage of the total number valid of rate constant patterns. We also checked what was the overlap between the patterns found when the complete dataset was used to fit the enzyme model and when each of the incomplete datasets were used. We show these results in Figure 4.11. In most cases, generating a 100 model ensembles with a 100 models each seems to be enough to sample the space of equivalent optima, even if not all models have an ssld < 0.1 and are thus discarded. The exception is when the k_{cat} is removed from the fitting dataset, and over 80% of the fitted rate constant sets are different, i.e. only 20% are repeated. Yet, here each model ensemble contains less than 40 rate constant sets. Overall, for GAPD the overlap of the rate constant sets obtained when using the complete fitting dataset and when using each of the incomplete fitting datasets is quite high, at least 80%, except when the K_{eq} and the k_{cat} are removed. For the latter it might be a matter of generating more models with ssld < 0.1 for each ensemble, while for the former it might be that K_{eq} is very constraining, as suggested by the Shannon entropy results. Thus, when it is removed from the fitting dataset the fitness landscape changes significantly, which would also be consistent with the higher number of different patterns found when the K_{eq} is not part of the fitting dataset. We see the same trends for TALB, although the overlap between the different patterns found when using the complete dataset for fitting and using each of the incomplete datasets is generally lower. This can be either because: a) as a data point is removed from the fitting dataset the fitness landscape changes causing some optima to appear and others disappear, or b) because not enough models were generated. Yet, given that, for instance, when the K_m for F6P is removed the % of different patterns is below 10%, but the overlap with the complete dataset is lower than 40%, there might indeed be that the fitness landscape changes such that the optima change, which hints at a conflict between different data points. This hypothesis is also consistent with the results of the parameter scan in Figure 4.6, where we see that the lowest ssld for TALB is higher than for GAPD and ENO, suggesting also a conflict between different data points. Finally, it is interesting to note that, even though GAPD and TALB have the same number of rate constants, for GAPD the amount of allowed combinations of rate constant patterns is much lower than for TALB. Without including ΔK_b in the fitting dataset, for GAPD only less than 5% of the total number of rate constant patterns

are different, while for TALB there are still $\sim 40\%$. This might be because the fitting dataset for GAPD includes a dissociation constant for NAD, which together with the enzyme's K_{eq} becomes quite constraining.



(a) GAPD



(b) TALB

Figure 4.11: Relative Shannon entropy results over 100 model ensembles for each enzyme, however, here each model ensemble contains only 30 models for GAPD and TALB, and 50 models for ENO. Each box represents the relative Shannon entropy when a given data point is removed from the fitting dataset. In red all data points were used to fit the enzyme, in blue ΔK_b was removed from the fitting dataset, and so on.

ΔK_b impact on enzyme kinetic model dynamics

Finally, we check if the integration of ΔK_b affects the enzyme-level kinetic model dynamics. Since including the ΔK_b in the fitting dataset has a greater impact on

ENO, in Figure 4.12 we show the respective metabolite time courses obtained from simulations in a closed system with no perturbations. Metabolite concentrations are simply expected to evolve until equilibrium is reached. We simulate only one model ensemble, however, the results are similar for all 100 ensembles. Each line in the plots represent one model. When ΔK_b is included in the fitting dataset the system dynamics are visibly more constrained, which is expected given that the rate constant set patterns allowed are also drastically reduced. Since ENO is an extreme case, in Figure 4.13 we show the metabolite time-courses for TALB. The impact of ΔK_b on TALB's system dynamics seems to be almost negligible. The same happens for GAPD. Thus, we conclude that only when the number of rate constant sets is dramatically constrained is there a visible impact on the metabolite time-courses at the enzyme-level.

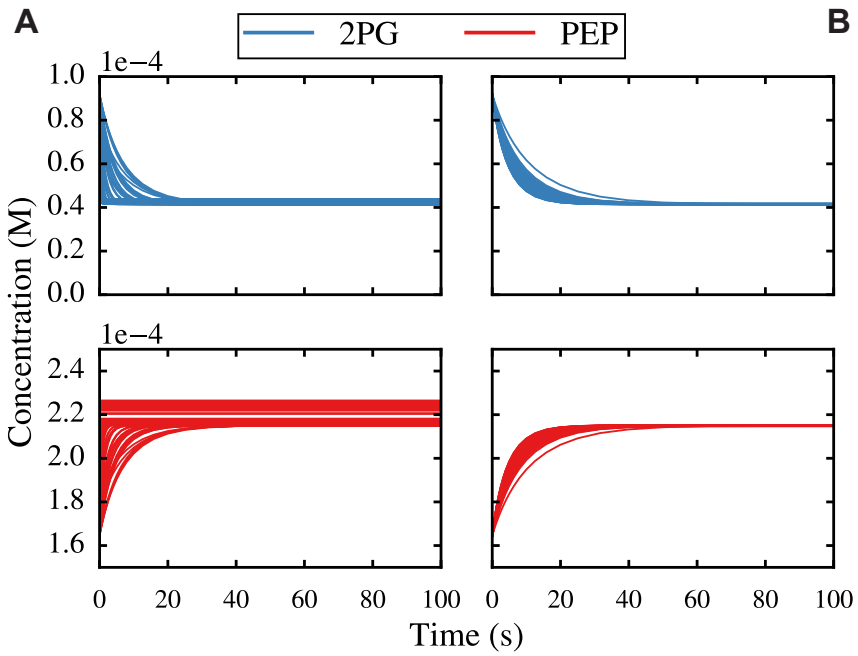


Figure 4.12: Metabolite time course results for one ENO model ensemble when (A) ΔK_b was not included in the fitting dataset, and (B) when it was included in the fitting dataset. Each line represents one model.

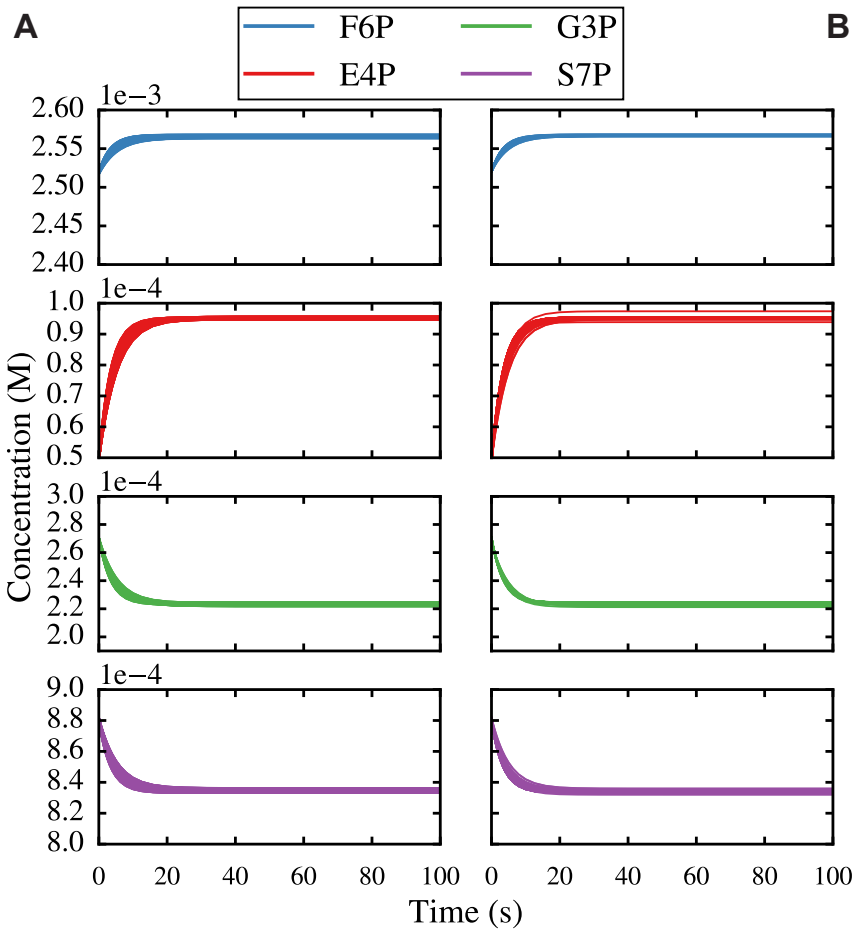


Figure 4.13: Metabolite time course results for one TALB model ensemble when (A) ΔK_b was not included in the fitting dataset, and (B) when it was included in the fitting dataset. Each line represents one model.

4.5 Conclusions

In this work we used MM-PBSA to calculate relative binding energies between substrate(s) and product(s) for three wild-type enzymes in the central carbon metabolism of *E. coli*. For ENO we constrained the ratio of four out of six rate constants, for GAPDH we constrained the ratio of eight out of 12 rate constants, and for TALB we constrained the ratios of four out of 12 rate constants, thus being able to test the impact of different degrees of constraining through ΔK_b. Regarding the reduction of valid rate constant combinations, we find that ΔK_b is among the most powerful parameters. Furthermore, when the ΔK_b value is on the limit of compatibility with the remaining data points, the amount of valid rate constant combinations becomes extremely reduced, with a significant impact on the respective enzyme-level dynamic

Enzyme	Crystal structure PDB
ENO	1E9I, subunit D [33]
ENO	3H8A, subunit A [45]
GAPD	1GAD [11]
TALB	1UCW [29]

Table 4.2: Crystal structures used for each enzyme.

behavior, which is otherwise barely visible. The similar impact of ΔK_b in the amount of valid rate constant combinations in GAPD and TALB, despite the different proportion of rate constants constrained by the ΔK_b, might be because GAPD is already more tightly constrained before including the ΔK_b in the fitting dataset than TALB.

One obstacle to scale-up this approach to more enzymes in the metabolism seem to be metalloenzymes such as Enolase. Even though we didn't show it here, we also applied this workflow to ADK1 which, like ENO, has two Mg^{2+} ions bound, and the variance in ΔG was very high. Therefore we cannot simply apply an automatic approach on metalloezymes, some steps may need to be done manually, e.g. docking of ligands and/or metal ions. Other challenging enzymes could be allosteric enzymes such as PFK or PYK, as it is hard to model the effect that effector molecule binding has on enzyme conformation with molecular mechanics methods.

Therefore, we conclude that 1) non-allosteric and non metalloenzymes are the best targets to apply the workflow presented here; and 2) ΔK_b can be as constraining as any other kinetic/thermodynamic parameter and is definitely worth integrating it with metabolic kinetic models.

Summing up, this is a promising workflow not only to parameterize metabolic kinetic models, but also to evaluate enzyme substrate promiscuity and to take into account enzyme mutations in kinetic models. Besides, it can be integrated with any type of kinetic model that relies on elementary rate constants [64, 54, 53].

4.6 Methods

4.6.1 Enzyme crystal structures

See table 4.2 for the crystal structures used for the molecular dynamics simulations. For the MD simulation of GAPD with both NAD and G3P bound, G3P was

bound manually and the coordinates were taken from the structure 1DC4 [71]. For the MD simulations of TALB with S7P bound, S7P was bound manually.

4.6.2 Ligand parameterization

The following ligands bound or to be docked to the enzyme were parameterized using Gaussian 09 [13] in the gas phase at the HF/6-31G* level: 2PG, PEP, 13DPG, G3P, F6P, S7P.The ligand charges were fitted using the RESP technique [5]. Antechamber [65] was then used to assign charges and atom types for compatibility with the MD simulations using the AMBER ff99SB force field. The ligand structures were obtained from PDBeChem [10, 19], where the ideal structures were used. The parameters for the cofactors NAD^+, NADH, and Mg^{2+} were obtained from the AMBER parameter database by the Bryce group [1].

4.6.3 Molecular dynamics simulations

The PMEMD module from the AMBER14 toolkit [7] is used to run all MD simulations. Each simulation is preceded by a minimization step at constant volume for up to 10^6 cycles, and an equilibration step at constant volume where the system is heated from 50 K to 300 K from step=0 to step=40000, and from step=40000 to step=500.000 temperature is kept constant at 300 K, while the enzyme and bound ligands are kept restrained. MD production simulations start off from the equilibration step, and are done at constant pressure. The Langevin thermostat is used for the equilibration and production runs, and the Particle Mesh Ewald (PME) cutoff is set to 12 Å. This procedure is used for all MD simulations, where the only variable is the length of the production simulation.

To generate topologies, the force field ff99SB was used, after which the system was solvated in a TIP3P 12 Å periodic water box, where 12 Å is the minimum distance between any atom in the enzyme/ligand and the periodic box boundary. Counterions were added, either Na^+ or Cl^-. MD simulations are performed in two different phases. In phase one the enzyme is simulated for at least 80 ns, to sample different enzyme conformations for docking. From each simulation we take every 100^{th} frame. See table 4.3 for details on the simulations.

Enzyme	Cofactors/ligands bound	Simulation time	Number of frames extracted for docking
ENO_1E9I	Mg^{2+}	110 ns	750
ENO_3H8A	2 Mg^{2+}, 2PG	75 ns	252
ENO_3H8A	2 Mg^{2+}, PEP	75 ns	252
GAPDH	NAD^{+}	110 ns	750
GAPDH	NAD^{+}, G3P	110 ns	745
TALB	-	83.6 ns	803
TALB	S7P	59.5 ns	493

Table 4.3: For each MD simulation, which enzyme was simulated, which cofactors/ligands were bound, how long was the simulation (in ns), and how many frames were extracted from each simulation for docking.

In phase two, MD simulations are run for 1 ns for all enzyme-ligand complexes: ENO_3H8A-2PG, ENO_3H8A-PEP, GAPDH-NAD-G3P, GAPDH-NAD-13DPG, TALB-S7P, and TALB-F6P. The results from these simulations are then used to estimate the ligand-enzyme binding energies with MM-PBSA.

Because force fields in general cannot model the interactions of metal ions with the enzyme/ligand properly, in all ENO's simulations, we restricted the position of the Mg^{2+} atoms by defining distances restraints to nearby atoms.

4.6.4 Docking

From the first MD simulation, we first remove any ligands that were bound (but not cofactors), then we load it on Chimera [49] using the MD Movie function and use Dock Prep on it to extract each frame and prepare it for docking. We use DOCK6 [35] to flexibly dock the ligands to the respective enzyme, which is considered as a single rigid structure. DOCK6 was used with the default parameters and the binding residues were defined either based on literature or the ligand's surrounding residues in the respective crystal structure, see table 4.4. From this step, we get > 250 enzyme-ligand complex frames which are then clustered according to the euclidean distances for three atom pairs, where one atom is part of the ligand, and the other is part of the enzyme. To do so we use the mean shift method implemented in scikit-learn [48]. The atom pairs to use are chosen based on the crystal structure, where the atoms in the enzyme are chosen as the closest ones to the respective pair on the ligand. From each of the five to ten largest clusters, we generate a set of representative frames, for which each atom pair distance is within the respective cluster's $mean \pm std$. From this

group of representative frames we choose n frames randomly (unless a frame looks like an outlier) to run the second phase of MD simulations. See table 4.5 for details on how many frames per enzyme-ligand complex were chosen as starting points for the second phase MD simulation (MD2), and how many of these MD simulations were considered to calculate the final $\Delta\Delta G$ value.

Enzyme	Binding residues
ENO	Lys341, Lys392 Arg370, Glu208, Ser371, Glu167, His369, Ser368, Leu339
GAPDH	Cys149, Ser148, Thr150, Thr208, Gly209, Arg231, His17
TALB	Asp17, Asn35, Lys132, Asn154, Ser176, Arg181, Ser226, Arg228, Thr242, Thr32, Ser93

Table 4.4: Binding residues used to dock ligands to the respective enzyme.

Enzyme-ligand complexes	Enzyme	Cofactors/ligands bound in MD1	Number of starting frames for MD2	Number of MD2 simulations used to calculate the final $\Delta\Delta G$
ENO-Mg^{2+}-2PG	ENO_1E9I	2 Mg^{2+}	12	-
ENO-Mg^{2+}-PEP	ENO_1E9I	2 Mg^{2+}	12	-
ENO-2Mg^{2+}-2PG	ENO_3H8A	2 Mg^{2+}, 2PG	50	50
ENO-2Mg^{2+}-PEP	ENO_3H8A	2 Mg^{2+}, PEP	67	43
GAPD-NAD-G3P	GAPDH	NAD^+	25	25
GAPD-NAD-13DPG	GAPDH	NAD^+	25	25
GAPD-NAD-G3P	GAPDH	NAD^+, G3P	25	25
GAPD-NAD-13DPG	GAPDH	NAD^+, G3P	25	25
TALB-F6P	TALB	-	25	20
TALB-S7P	TALB	-	25	20
TALB-F6P	TALB	S7P	25	25
TALB-S7P	TALB	S7P	25	20

Table 4.5: For each enzyme-ligand complex, which enzyme is part of it, which cofactors/ligands were bound to the enzyme in the first MD simulation (used to generate the docking conformations), how many starting frames for the second MD simulation were extracted, and how many MD2 simulations were considered to calculate the final $\Delta\Delta G$ value.

4.6.5 MM-PBSA

MMPBSA.py [42] from the AMBER14 toolkit was used to run the MM-PBSA calculations using the single simulation approach. We did not estimate the entropy term. To calculate the binding energies for a given enzyme and ligand, the MD simulations from phase two are used, in which the ligand of interest is bound to the enzyme. The first 0.1 ns of the simulation was always discarded, as we assume the enzyme-ligand complex may still be stabilizing. A total of 62 frames were used as input to the MMPBSA.py script for the 1 ns simulations. The bondi radii set was used by AMBER to generate the parameter/topology sets. The parameters for MM-PBSA were defined with a SASA-based model (inp=1) for non-polar solvation free energies calculations.

4.6.6 Enzyme-level metabolic kinetic models

To build each enzyme kinetic model, we use MASSef, presented in chapter 2 a Mathematica® software package that implements the eMASS framework. Therefore, we first collect kinetic data, such as k_{cat} or K_m, from literature and relevant databases, e.g. Brenda [51], Sabio-RK [68], or Biocyc [30]. Given an enzyme mechanism, which is assumed to be random if no information is found in the literature, the biochemical reaction is divided into each elementary step. Each reaction elementary step is modeled by a mass action kinetic rate law, which results in several rate constants whose values are fitted to the macroscopic kinetic data obtained from literature. The data points included in the dataset used for enzyme fitting are, in general, different for each enzyme, the only data point available for every enzyme being the equilibrium constant, obtained from eQuilibrator [12], which estimates equilibrium constants using the component contribution framework [44]. The complete fitting dataset for ENO includes the K_{eq}, a K_m for the reaction substrate 2PG, the ΔK_b estimated from MD simulations, and a k_{cat} whose value was set as the maximum between PYK's and PGM's k_{cat} - a k_{cat} is needed for the metabolite concentration changes to reach equilibrium in a reasonable time frame, i.e. 100 s rather than 10^7 s. GAPD's complete fitting dataset includes, besides the respective K_{eq} and ΔK_b, a K_m for each of the reaction substrates: G3P, PI, and NAD, a dissociation constant, K_d, for NAD, and a k_{cat} for the forward direction. TALB's complete fitting dataset includes the K_{eq}, ΔK_b, a K_m for each reaction substrate and product, two forward k_{cat}, and a K_i for phosphate inhibition with regards to all products and substrates. To fit each enzyme's rate constants to the respective kinetic data, a particle swarm

optimization algorithm is used to generate a set of 100 initial solutions, which are further optimized by the Levenberg-Marquardt algorithm, resulting in one model ensemble with a 100 enzyme-level models. We then repeat the procedure a 100 times to generate 100 model ensembles, with a 100 models each. These models' fitness is evaluated according to

$$ssld_{model} = \sum_{i=0}^{n} (\log(x_{data_i}) - log(x_{predicted_i}))^2$$

If the model *ssld* is greater than 0.1 the model is discarded, hence some model ensembles might have less than a 100 models. In this case an ssld < 0.1 means that the maximum error associated with the prediction of the parameters used to fit the elementary rate constants is less than 10% with some outliers below 40%. The kinetic data used to fit each enzyme is available in the github repository, as well as the Mathematica® notebooks used to generate the fitting data and treat the fit results. For more details about building the enzyme-level kinetic models, please see chapter 2.

4.6.7 Rate constant analysis

To plot the cluster tree and heatmap in Figure 4.8 we used the Python package seaborn 0.7.0 [67]. This cluster tree represents the hierarchical clustering of each model ensemble by rate constants. Only valid models, i.e. which satisfy $ssld_{model} < 0.1$ are considered in this step.

4.6.8 Model dynamics analysis

To analyze each enzyme-level model dynamics, we need to first convert the kinetic models made solely of rate constants into a model that includes also initial enzyme form and metabolite concentrations, so that the system dynamics can actually be simulated. We get the initial total metabolite concentrations from [46], while the initial enzyme form (i.e. free enzyme and enzyme-ligand complexes) concentrations are calculated by requiring the total enzyme concentration to be equal to the sum of all enzyme form concentrations and the variation in enzyme form concentration to be zero when the system is at steady-state. The total enzyme concentrations are taken from the proteomics dataset in [28], and the molecular weights in Biocyc [30] are used to convert the enzyme concentrations to units of M. Once the

enzyme form concentrations have been calculated, the metabolite concentrations in the system are updated by subtracting the amount of metabolite bound to enzyme. The Mathematica® notebooks used to build such models can be downloaded from https://github.com/martamatos/eMASS-MD**.** Once all models are in a format that can be simulated, the MASS toolbox [56], a Mathematica® package, is used to simulate each model for a 100 s, which was determined by visual inspection to be long enough for the system to reach equilibrium, i.e $K_{eq} = \sum_i [product_i] / \sum_i [substrate_i]$ and the flux through each elementary reaction is null. Only valid models are considered, i.e. which satisfy $ssld_{model} < 0.1$ and whose initial metabolite concentrations are non-negative.

4.7 Supplementary information

4.7.1 ΔG values

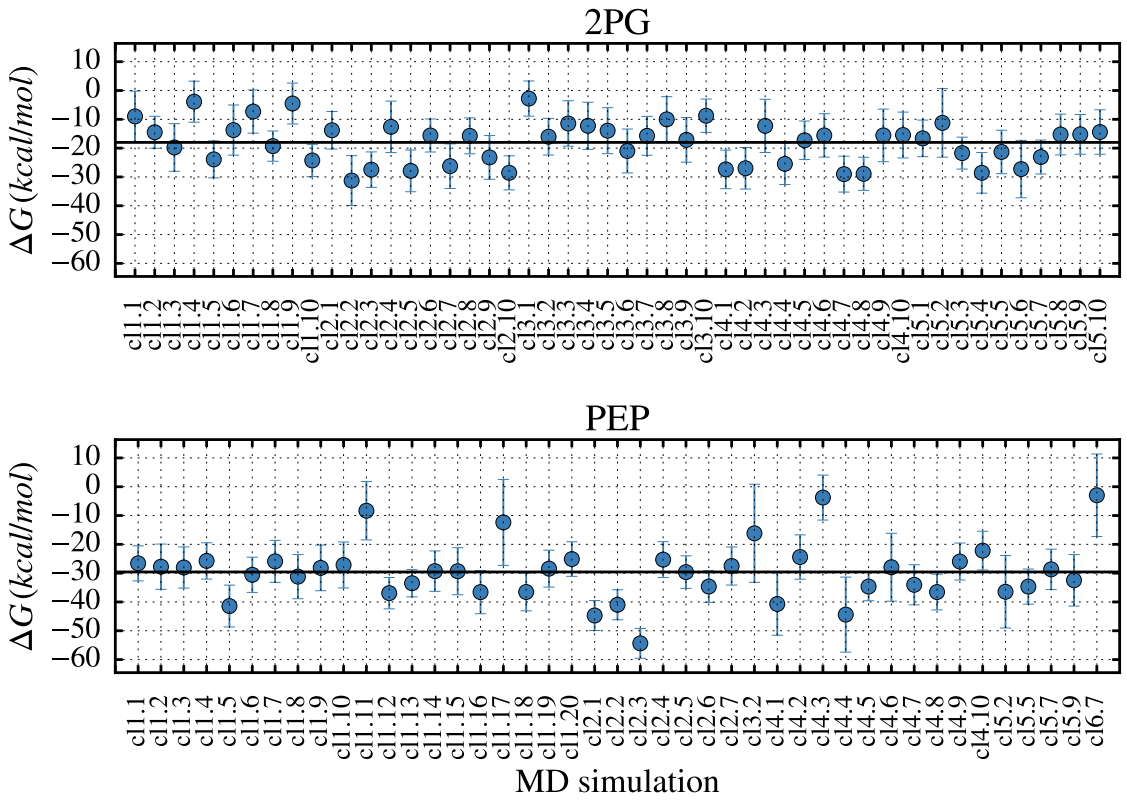


Figure 4.14: Binding energy,ΔG, values calculated from different molecular dynamics simulations for complexes formed by ENO and respective substrate (2PG, top) and product (PEP, bottom). clx.y represents the MD simulations for which starting frame y was taken from cluster x. The black line denotes the mean ΔG value for the respective complex. Error bars correspond to the standard deviation associated to the ΔG values obtained from each simulation.

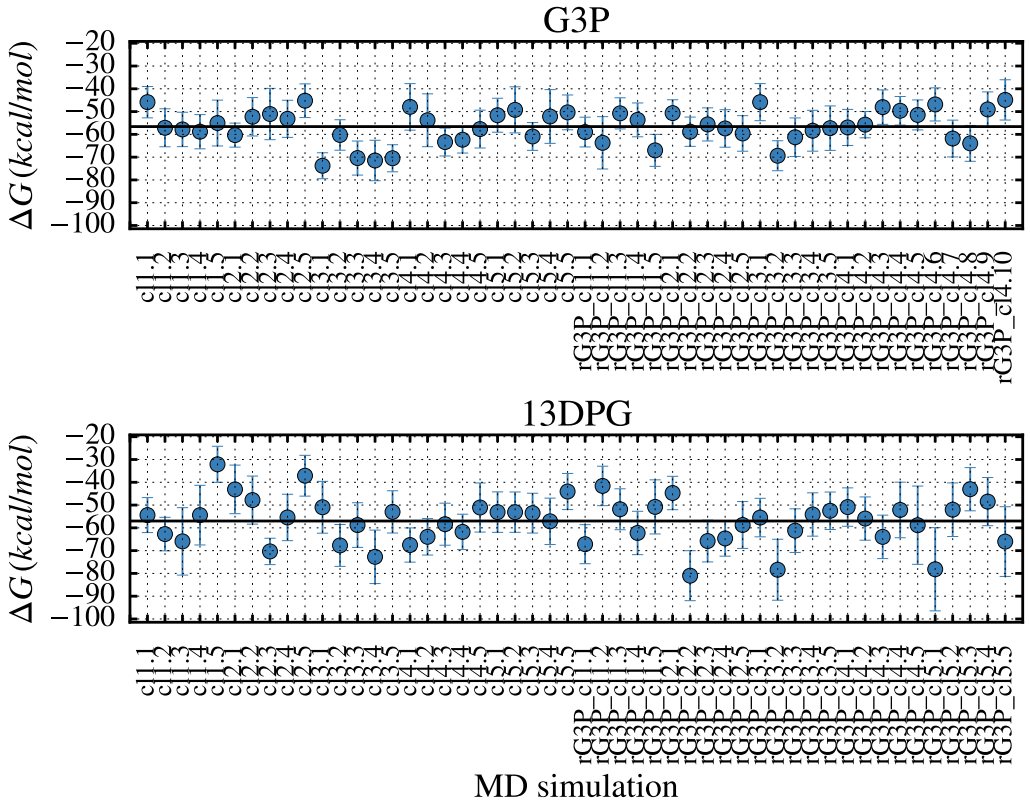


Figure 4.15: Binding energy,ΔG, values calculated from different molecular dynamics simulations for complexes formed by GAPD and respective substrates (NAD and G3P, top) and products (NADH and 13DPG, bottom). clx.y represents the MD simulations for which starting frame y was taken from cluster x. The prefix rG3P means the starting frame was obtained from the simulation of GAPD's holo form with both NAD and G3P bound, where G3P was removed before docking. The black line denotes the mean ΔG value for the respective complex. Error bars correspond to the standard deviation associated to the ΔG values obtained from each simulation.

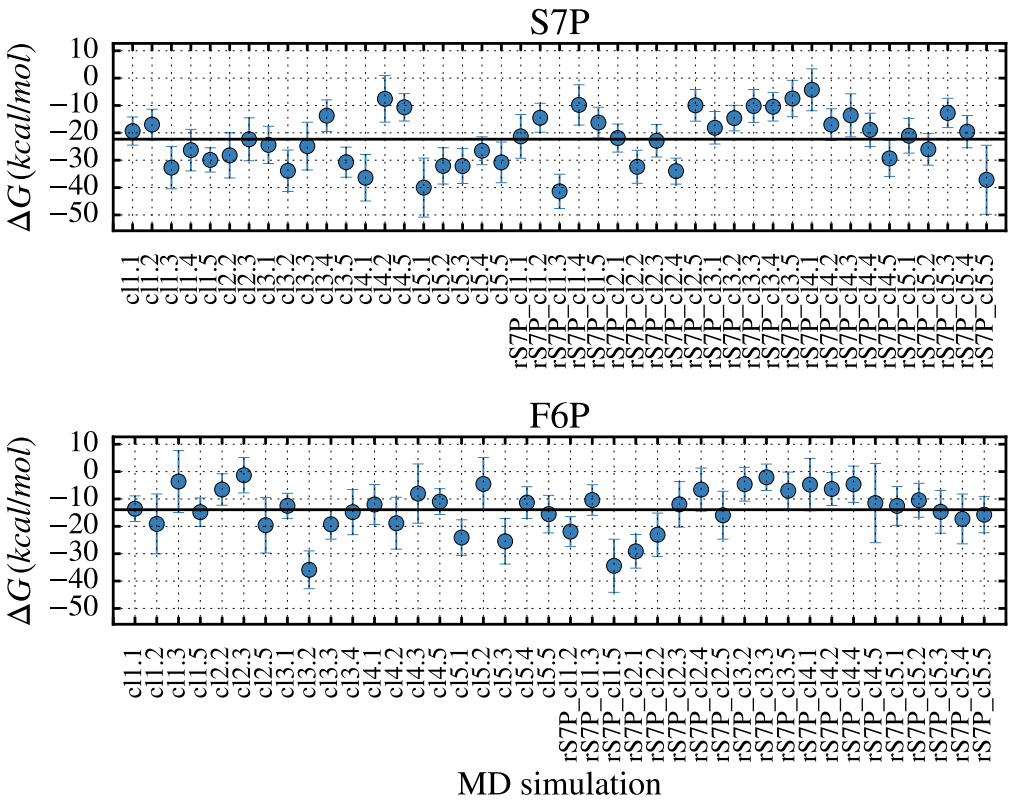


Figure 4.16: Binding energy,ΔG, values calculated from different molecular dynamics simulations for complexes formed by TALB and respective substrate (S7P, top) and product (F6P, bottom). clx.y represents the MD simulations for which starting frame y was taken from cluster x. The prefix rS7P means the starting frame was obtained from the simulation of TALB holo form with S7P bound, where S7P was removed before docking. The black line denotes the mean ΔG value for the respective complex. Error bars correspond to the standard deviation associated to the ΔG values obtained from each simulation.

4.7.2 Conversion of $\Delta\Delta G$ into ΔK_b

ENO

At equilibrium, we have:

$$\overrightarrow{k_1}[ENO][2PG] = \overleftarrow{k_1}[ENO\&2PG]$$

$$\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}} = \frac{[ENO\&2PG]}{[ENO][2PG]}$$

$$\overrightarrow{k_2}[ENO\&PEP] = \overleftarrow{k_2}[ENO][PEP]$$
$$\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}} = \frac{[ENO\&PEP]}{[ENO][PEP]}$$

Calculating the difference in binding Gibbs energies:

$$\begin{aligned}
\Delta G_{PEP} - \Delta G_{2PG} &= -RT\,ln\left(\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}}\right) + RT\,ln\left(\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}}\right) \\
&= RT\,ln\left(\frac{\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}}}{\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}}}\right) \\
&= RT\,ln\left(\frac{\overrightarrow{k_1}\overrightarrow{k_2}}{\overleftarrow{k_1}\overleftarrow{k_2}}\right) \\
\frac{\overrightarrow{k_1}\overrightarrow{k_2}}{\overleftarrow{k_1}\overleftarrow{k_2}} &= \exp\left(\frac{\Delta G_{PEP} - \Delta G_{2PG}}{RT}\right)
\end{aligned}$$

GAPD

At equilibrium, we have:

$$\overrightarrow{k_1}[GAPD][NAD] = \overleftarrow{k_1}[GAPD\&NAD]$$
$$\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}} = \frac{[GAPD\&NAD]}{[GAPD][NAD]}$$

$$\overrightarrow{k_3}[GAPD\&NAD][G3P] = \overleftarrow{k_3}[GAPD\&NAD\&G3P]$$
$$\frac{\overrightarrow{k_3}}{\overleftarrow{k_3}} = \frac{[GAPD\&NAD\&G3P]}{[GAPD\&NAD][G3P]}$$

$$\overrightarrow{k_2}[GAPD\&NADH] = \overleftarrow{k_2}[GAPD][NADH]$$
$$\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}} = \frac{[GAPD\&NADH]}{[GAPD][NADH]}$$

$$\overrightarrow{k_4}[GAPD\&NADH\&DPG] = \overleftarrow{k_4}[GAPD\&NADH][DPG]$$
$$\frac{\overleftarrow{k_4}}{\overrightarrow{k_4}} = \frac{[GAPD\&NADH\&DPG]}{[GAPD\&NADH][DPG]}$$

Calculating the difference in binding Gibbs energies:

$$\begin{aligned}
\Delta G_{NADH\&DPG} - \Delta G_{NAD\&G3P} &= -RL\,ln\left(\frac{\overleftarrow{k_2}\,\overleftarrow{k_4}}{\overrightarrow{k_2}\,\overrightarrow{k_4}}\right) + RT\,ln\left(\frac{\overrightarrow{k_1}\,\overrightarrow{k_3}}{\overleftarrow{k_1}\,\overleftarrow{k_3}}\right) \\
&= RT\,ln\left(\frac{\frac{\overrightarrow{k_1}\,\overrightarrow{k_3}}{\overleftarrow{k_1}\,\overleftarrow{k_3}}}{\frac{\overleftarrow{k_2}\,\overleftarrow{k_4}}{\overrightarrow{k_2}\,\overrightarrow{k_4}}}\right) \\
&= RT\,ln\left(\frac{\overrightarrow{k_1}\overrightarrow{k_3}\overrightarrow{k_2}\overrightarrow{k_4}}{\overleftarrow{k_1}\overleftarrow{k_3}\overleftarrow{k_2}\overleftarrow{k_4}}\right) \\
\frac{\overrightarrow{k_1}\overrightarrow{k_3}\overrightarrow{k_2}\overrightarrow{k_4}}{\overleftarrow{k_1}\overleftarrow{k_3}\overleftarrow{k_2}\overleftarrow{k_4}} &= \exp\left(\frac{\Delta G_{NADH\&DPG} - \Delta G_{NAD\&G3P}}{RT}\right)
\end{aligned}$$

TALB

At equilibrium, we have:

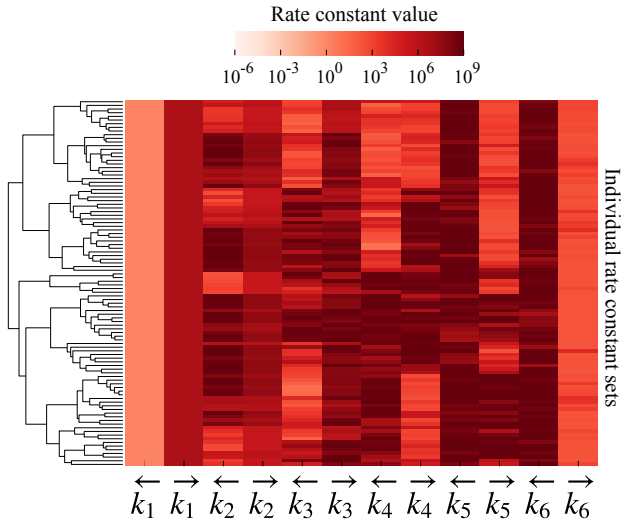
$$\overrightarrow{k_1}[TALB][S7P] = \overleftarrow{k_1}[TALB\&S7P]$$
$$\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}} = \frac{[TALB\&S7P]}{[TALB][S7P]}$$

$$\overrightarrow{k_2}[TALB\&F6P] = \overleftarrow{k_2}[TALB][F6P]$$
$$\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}} = \frac{[TALB\&F6P]}{[TALB][F6P]}$$

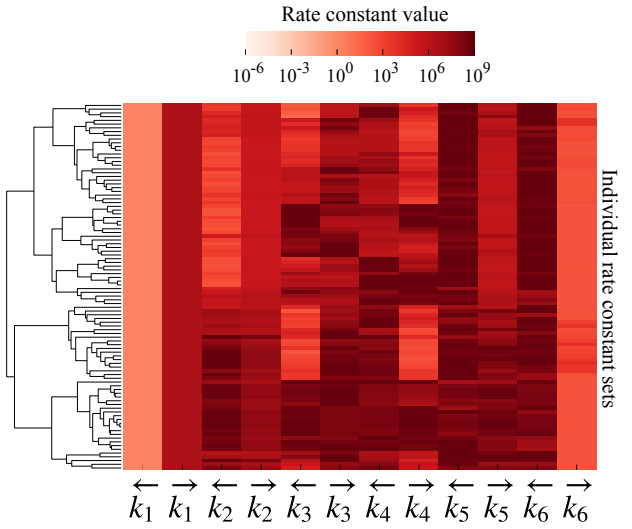
Calculating the difference in binding Gibbs energies:

$$\begin{aligned}
\Delta G_{F6P} - \Delta G_{S7P} &= -RT\,ln\left(\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}}\right) + RT\,ln\left(\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}}\right) \\
&= RT\,ln\left(\frac{\frac{\overrightarrow{k_1}}{\overleftarrow{k_1}}}{\frac{\overleftarrow{k_2}}{\overrightarrow{k_2}}}\right) \\
&= RT\,ln\left(\frac{\overrightarrow{k_1}\overrightarrow{k_2}}{\overleftarrow{k_1}\overleftarrow{k_2}}\right) \\
\frac{\overrightarrow{k_1}\overrightarrow{k_2}}{\overleftarrow{k_1}\overleftarrow{k_2}} &= \exp\left(\frac{\Delta G_{F6P} - \Delta G_{S7P}}{RT}\right)
\end{aligned}$$

4.7.3 Clustermaps for GAPD and TALB

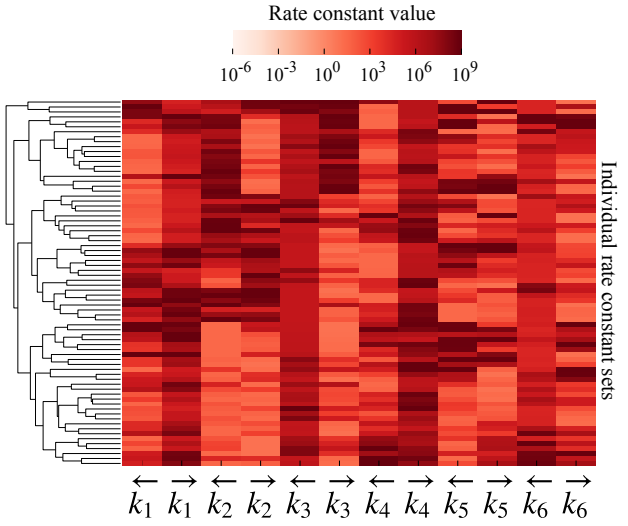


(a)

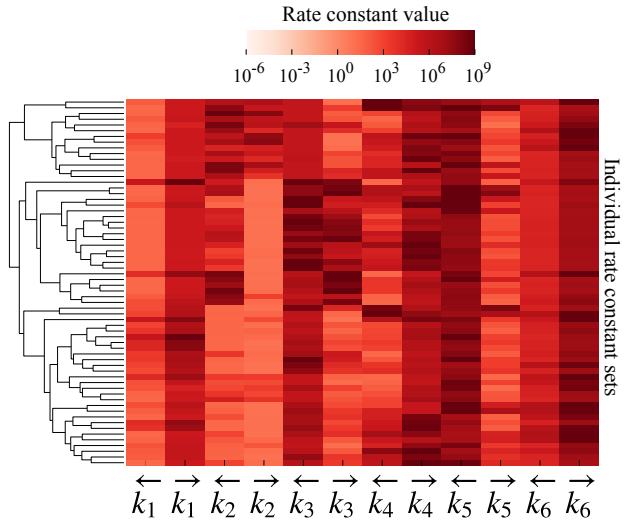


(b)

Figure 4.17: Example of the results for the clustering of rate constant sets for GAPD when (a) ΔK_b is not included in the fitting dataset, and (b) when $\Delta K_b$is included in the fitting dataset.



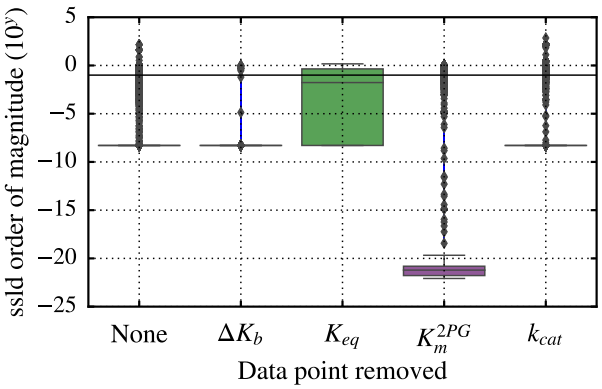
(a) Without ΔK_b



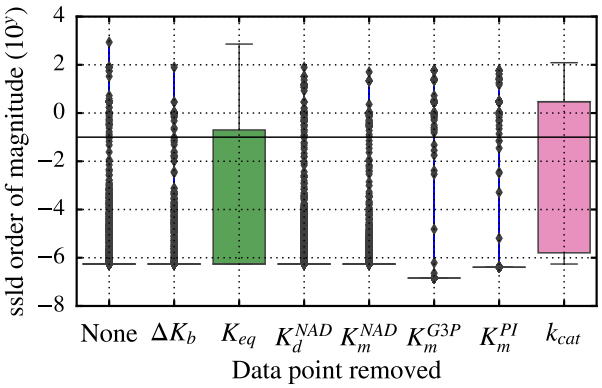
(b) With ΔK_b

Figure 4.18: Example of the results for the clustering of rate constant sets for TALB when (a) ΔK_b is not included in the fitting dataset, and (b) when $\Delta K_b$is included in the fitting dataset.

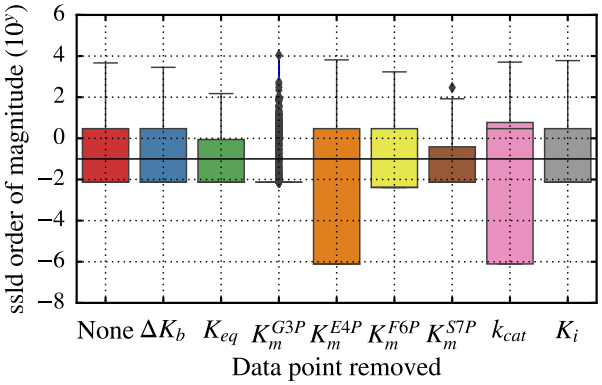
4.7.4 Parameter impact on model fitness



(a) ENO



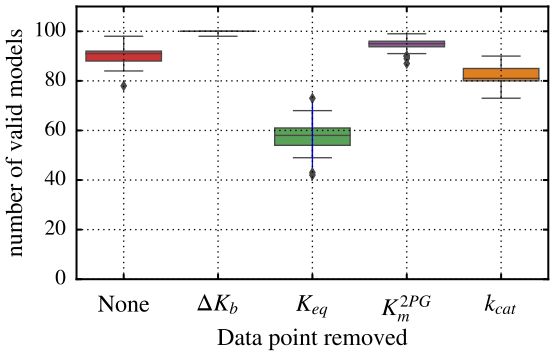
(b) GAPD



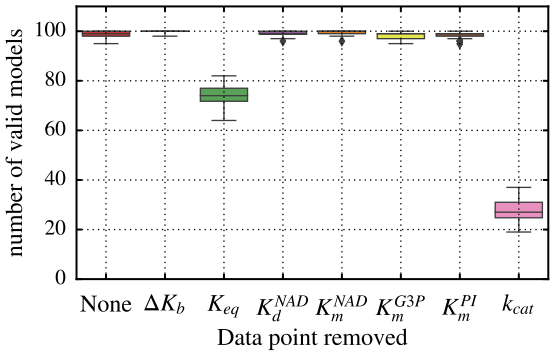
(c) TALB

Figure 4.19: Logarithm of the sum of squared errors for each model when a given data point was removed from the fitting dataset. Fits with ssld < 0.1 (horizontal black line) are considered to be good fits.

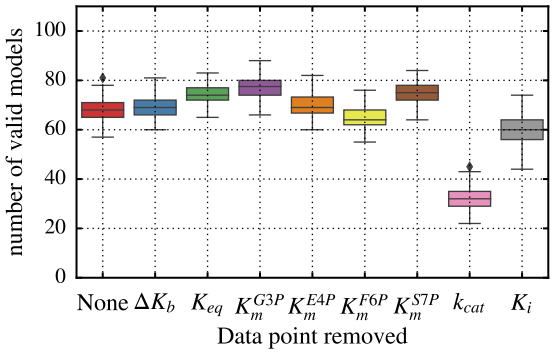
4.7.5 Parameter impact on the amount of valid models per ensemble



(a) ENO



(b) GAPD



(c) TALB

Figure 4.20: Number of models with ssld< 0.1 in each model ensemble l when a given data point was removed from the fitting dataset.

References

[1] URL: http://research.bmh.manchester.ac.uk/bryce/amber.

[2] J. Almquist, M. Cvijovic, V. Hatzimanikatis, J. Nielsen, and M. Jirstrand. Kinetic models in industrial biotechnology - Improving cell factory performance. *Metabolic engineering*, 24C:38–60, 2014.

[3] J. Åqvist, C. Medina, and J.-E. Samuelsson. A new method for predicting binding affinity in computer-aided drug design. *Protein engineering, design and selection*, 7(3):385–391, 1994.

[4] X. Barril, J. L. Gelpí, J. M. López, M. Orozco, and F. J. Luque. How accurate can molecular dynamics/linear response and Poisson–Boltzmann/solvent accessible surface calculations be for predicting relative binding affinities? Acetylcholinesterase huprine inhibitors as a test case. *Theoretical chemistry accounts*, 106(1):2–9, June 1, 2001.

[5] C. I. Bayly, P. Cieplak, W. Cornell, and P. A. Kollman. A well-behaved electrostatic potential based method using charge restraints for deriving atomic charges: the RESP model. *The journal of physical chemistry*, 97(40):10269–10280, 1993.

[6] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[7] D. Case, V. Babin, J. Berryman, R. Betz, Q. Cai, and e. a. DS Cerutti. Amber14. University of California.

[8] R. S. Costa, A. Hartmann, and S. Vinga. Kinetic modeling of cell metabolism for microbial production. *Journal of biotechnology*, 219:126–141, 2015.

[9] M. De Vivo, M. Masetti, G. Bottegoni, and A. Cavalli. Role of molecular dynamics and related methods in drug discovery. eng. *Journal of medicinal chemistry*, 59(9):4035–4061, 2016. ISSN: 15204804, 00222623. DOI: 10.1021/acs.jmedchem.5b01684.

[10] D. Dimitropoulos, J. Ionides, and K. Henrick. *Using MSDchem to Search the PDB Ligand Dictionary*. In. *Current protocols in bioinformatics*. John Wiley & Sons, Inc., 2002. ISBN: 9780471250951.

[11] E. Duée, L. Olivier-Deyris, E. Fanchon, C. Corbier, G. Branlant, and O. Dideberg. Comparison of the structures of wild-type and a N313T mutant of *Escherichia coli* glyceraldehyde 3-phosphate dehydrogenases: implication for NAD binding and cooperativity. *Journal of molecular biology*, 257:814–838, 1996.

[12] A. Flamholz, E. Noor, A. Bar-Even, and R. Milo. EQuilibrator - The biochemical thermodynamics calculator. *Nucleic acids research*, 40(D1):770–775, 2012.

[13] M. Frisch, G. Trucks, H. Schlegel, G. Scuseria, M. Robb, and e. a. JR Cheeseman. Gaussian09. Gaussian, Inc., 2004.

[14] R. Gabdoulline, U. Kummer, L. Olsen, and R. Wade. Concerted simulations reveal how peroxidase compound iii formation results in cellular oscillations. eng. *Biophysical journal*, 85(3):1421–1428, 2003. ISSN: 15420086, 00063495. DOI: 10.1016/S0006-3495(03)74574-3.

[15] S. Genheden. MM/GBSA and LIE estimates of host–guest affinities: dependence on charges and solvation model. *Journal of computer-aided molecular design*, 25(11):1085–1093, Nov. 1, 2011.

[16] S. Genheden, I. Nilsson, and U. Ryde. Binding Affinities of Factor Xa Inhibitors Estimated by Thermodynamic Integration and MM/GBSA. *Journal of chemical information and modeling*, 51(4):947–958, 2011.

[17] S. Genheden and U. Ryde. The MM/PBSA and MM/GBSA methods to estimate ligand-binding affinities. *Expert opinion on drug discovery*, 10(5): 2015.

[18] S. Genheden and U. Ryde. Will molecular dynamics simulations of proteins ever reach equilibrium? *Physical chemistry chemical physics : pccp*, 14(24):8662–77, 2012.

[19] A. Golovin, D. Dimitropoulos, T. Oldfield, A. Rachedi, and K. Henrick. MSDsite: A database search and retrieval system for the analysis and viewing of bound ligands and active sites. *Proteins: structure, function, and bioinformatics*, 58(1):190–199, 2005.

[20] H. Gouda, I. D. Kuntz, D. A. Case, and P. A. Kollman. Free energy calculations for theophylline binding to an RNA aptamer: Comparison of MM-PBSA and thermodynamic integration methods. *Biopolymers*, 68(1):16–34, 2003.

[21] C. R. W. Guimarães. A Direct Comparison of the MM-GB/SA Scoring Procedure and Free-Energy Perturbation Calculations Using Carbonic Anhydrase as a Test Case: Strengths and Pitfalls of Each Approach. *Journal of chemical theory and computation*, 7(7):2296–2306, 2011.

[22] T. Hansson, J. Marelius, and J. Åqvist. Ligand binding affinity prediction by linear interaction energy methods. *Journal of computer-aided molecular design*, 12(1):27–35, Jan. 1, 1998.

[23] B.-J. Harder, K. Bettenbrock, and S. Klamt. Model-Based Metabolic Engineering Enables High Yield Itaconic Acid Production by *Escherichia coli*. *Metabolic engineering*, 38:29–37, 2016.

[24] T. Hou, S. Guo, and X. Xu. Predictions of binding of a diverse set of ligands to gelatinase-A by a combination of molecular dynamics and continuum solvent models. *Journal of physical chemistry b*, 106(21):5527–5535, 2002.

[25] T. Hou, J. Wang, Y. Li, and W. Wang. Assessing the Performance of the MM/PBSA and MM/GBSA Methods . 1 . The Accuracy of Binding Free Energy Calculations Based on Molecular Dynamics Simulations:69–82, 2011.

[26] P. A. K. I. Bea E. c. and C. jaime. Molecular Recognition by beta-Cyclodextrin Derivatives: FEP vs MM/PBSA Study. *Combinatorial chemistry & high throughput screening*, 4(8):605–611, 2001.

[27] A. W. I Muegge H. T. A fast estimate of electrostatic group contributions to the free energy of protein inhibitor binding. *Prot eng*, 10:1363–1372, 1998.

[28] N. Ishii, Y. Suga, A. Hagiya, H. Watanabe, H. Mori, M. Yoshino, and M. Tomita. Dynamic simulation of an *in vitro* multi-enzyme system. *Febs letters*, 581(3):413–420, 2007.

[29] J. Jia, U. Schörken, Y. Lindqvist, G. a. Sprenger, and G. Schneider. Crystal structure of the reduced Schiff-base intermediate complex of transaldolase B from *Escherichia coli*: mechanistic implications for class I aldolases. *Protein science : a publication of the protein society*, 6:119–124, 1997.

[30] I. M. Keseler, A. Mackie, M. Peralta-Gil, A. Santos-Zavaleta, S. Gama-Castro, C. Bonavides-Martínez, C. Fulcher, A. M. Huerta, A. Kothari, M. Krummenacker, M. Latendresse, L. Muñiz-Rascado, Q. Ong, S. Paley, I. Schröder, A. G. Shearer, P. Subhraveti, M. Travers, D. Weerasinghe, V. Weiss, J. Collado-Vides, R. P. Gunsalus, I. Paulsen, and P. D. Karp. EcoCyc: Fusing model organism databases with systems biology. *Nucleic acids research*, 41(D1):605–612, 2013.

[31] P. A. Kollman, I. Massova, C. Reyes, B. Kuhn, S. Huo, L. Chong, M. Lee, T. Lee, Y. Duan, W. Wang, O. Donini, P. Cieplak, J. Srinivasan, D. A. Case, and T. E. Cheatham. Calculating Structures and Free Energies of Complex Molecules: Combining Molecular Mechanics and Continuum Models. *Accounts of chemical research*, 33(12):889–897, 2000.

[32] B. Kuhn and P. A. Kollman. Binding of a Diverse Set of Ligands to Avidin and Streptavidin: An Accurate Quantitative Prediction of Their Relative Affinities by a Combination of Molecular Mechanics and Continuum Solvent Models. *Journal of medicinal chemistry*, 43(20):3786–3791, 2000.

[33] K. Kühnel and B. F. Luisi. Crystal structure of the *Escherichia coli* RNA degradosome component enolase. *Journal of molecular biology*, 313:583–592, 2001.

[34] T. Laitinen, J. A. Kankare, and M. Peräkylä. Free energy simulations and MM–PBSA analyses on the affinity and specificity of steroid binding to antiestradiol antibody. *Proteins: structure, function, and bioinformatics*, 55(1):34–43, 2004.

[35] P. T. Lang, S. R. Brozell, S. Mukherjee, E. F. Pettersen, E. C. Meng, V. Thomas, R. C. Rizzo, D. A. Case, T. L. James, and I. D. Kuntz. DOCK 6: Combining techniques to model RNA-small molecule complexes. eng. *Rna-a publication of the rna society*, 15(6):1219–1230, 2009.

[36] F. S. Lee, Z.-T. Chu, M. B. Bolger, and A. Warshel. Calculations of antibody-antigen interactions: microscopic and semi-microscopic evaluation of the free energies of binding of phosphorylcholine analogs to McPC603. *Protein engineering, design and selection*, 5(3):215–228, 1992.

[37] S. Y. Lee and H. U. Kim. Systems strategies for developing industrial microbial strains. *Nature biotechnology*, 33(10):1061–1072, 2015.

[38] J. D. Madura, Y. Nakajima, R. M. Hamilton, A. Wierzbicki, and A. Warshel. Calculations of the electrostatic free energy contributions to the binding free energy of sulfonamides to carbonic anhydrase. *Structural chemistry*, 7(2):131–138, Apr. 1, 1996.

[39] I. Massova and P. a. Kollman. Combined molecular mechanical and continuum solvent approach (mm-pbsa/gbsa) to predict ligand binding. *Perspectives in drug discovery and design*, 18(Hydrophobicity and Solvation in Drug Design, Pt. II):113–135, 2000. ISSN: 0928-2866. DOI: Doi10.1023/A:1008763014207.

[40] N. Mih, E. Brunk, A. Bordbar, and B. O. Palsson. A multi-scale computational platform to mechanistically assess the effect of genetic variation on drug responses in human erythrocyte metabolism. eng. *Plos computational biology*, 12(7):e1005039, e1005039, 2016.

[41] P. Mikulskis, S. Genheden, P. Rydberg, L. Sandberg, L. Olsen, and U. Ryde. Binding affinities in the SAMPL3 trypsin and host–guest blind tests estimated with the MM/PBSA and LIE methods. *Journal of computer-aided molecular design*, 26(5):527–541, May 1, 2012.

[42] B. R. Miller, T. D. McGee, J. M. Swails, N. Homeyer, H. Gohlke, and A. E. Roitberg. MMPBSA.py: An Efficient Program for End-State Free Energy Calculations. *Journal of chemical theory and computation*, 8(9):3314–3321, 2012.

[43] H. S. Muddana, C. D. Varnado, C. W. Bielawski, A. R. Urbach, L. Isaacs, M. T. Geballe, and M. K. Gilson. Blind prediction of host-guest binding affinities: a new SAMPL3 challenge. eng. *Journal of computer-aided molecular design*, 26(5):475–487, 2012.

[44] E. Noor, H. S. Haraldsdóttir, R. Milo, and R. M. T. Fleming. Consistent estimation of Gibbs energy using component contributions. *Plos computational biology*, 9(7):e1003098, Jan. 2013.

[45] S. Nurmohamed, A. R. McKay, C. V. Robinson, and B. F. Luisi. Molecular recognition between *Escherichia coli* enolase and ribonuclease E. *Acta crystallographica. section d, biological crystallography*, 66:1036–40, 2010.

[46] J. O. Park, S. A. Rubin, Y.-f. Xu, D. Amador-noguez, J. Fan, T. Shlomi, and J. D. Rabinowitz. Metabolite concentrations, fluxes and free energies imply efficient enzyme usage. *Nature chemical biology*, (May), 2016.

[47] D. A. Pearlman. Evaluating the Molecular Mechanics PoissonBoltzmann Surface Area Free Energy Method Using a Congeneric Series of Ligands to p38 MAP Kinase. *Journal of medicinal chemistry*, 48(24):7796–7807, 2005.

[48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of machine learning research*, 12:2825–2830, 2011.

[49] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng, and T. E. Ferrin. UCSF Chimera—A visualization system for exploratory research and analysis. *Journal of computational chemistry*, 25(13):1605–1612, 2004.

[50] M. Petukh, M. Li, and E. Alexov. Predicting Binding Free Energy Change Caused by Point Mutations with Knowledge-Modified MM/PBSA Method. *Plos computational biology*, 11(7):1–23, 2015.

[51] S. Placzek, I. Schomburg, A. Chang, L. Jeske, M. Ulbrich, J. Tillack, and D. Schomburg. BRENDA in 2017: New perspectives and new tools in BRENDA. *Nucleic acids research*, 45(D1):D380–D388, 2017.

[52] N. D. Price, J. L. Reed, and B. Ø. Palsson. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nature reviews. microbiology*, 2(11):886–97, Nov. 2004.

[53] P. A. Saa and L. K. Nielsen. Construction of feasible and accurate kinetic models of metabolism: A Bayesian approach. *Scientific reports*, 6(July):29635, 2016.

[54] P. Saa and L. K. Nielsen. A general framework for thermodynamically consistent parameterization and efficient sampling of enzymatic reactions. *Plos computational biology*, 11(4):e1004195, 2015.

[55] M. Salvalaglio, L. Zamolo, V. Busini, D. Moscatelli, and C. Cavallotti. Molecular modeling of Protein A affinity chromatography. *Journal of chromatography a*, 1216(50):8678–8686, 2009. 12th Symposium on Preparative and Industrial Chromatography and Allied Techniques.

[56] A. Sastry and N. Sonnenschein. MASS-Toolbox: Introducing Annotations and Pathway Attributes. Jan. 2016. URL: `https://doi.org/10.5281/zenodo.44339`.

[57] C. N. Schutz and A. Warshel. What are the dielectric "constants" of proteins and how to validate electrostatic models? *Proteins: structure, function, and bioinformatics*, 44(4):400–417, 2001.

[58] Y. Y. Sham, Z. T. Chu, H. Tao, and A. Warshel. Examining methods for calculations of binding free energies: LRA, LIE, PDLD-LRA, and PDLD/S-LRA calculations of ligands binding to an HIV protease. *Proteins: structure, function, and bioinformatics*, 39(4):393–407, 2000.

[59] C. E. Shannon. A Mathematical Theory of Communication. *Bell system technical journal*, 27(3):379–423, 1948.

[60] M. R. Shirts, D. L. Mobley, and J. D. Chodera. Chapter 4 Alchemical Free Energy Calculations: Ready for Prime Time? *Annual reports in computational chemistry*, 3(07):41–59, 2007.

[61] J. Srinivasan, T. E. Cheatham, P. Cieplak, P. A. Kollman, and D. A. Case. Continuum Solvent Studies of the Stability of DNA, RNA, and PhosphoramidateDNA Helices. *Journal of the american chemical society*, 120(37):9401–9409, 1998.

[62] M. Stein, R. R. Gabdoulline, and R. C. Wade. Bridging from molecular simulation to biochemical networks. eng. *Current opinion in structural biology*, 17(2):166–172, 2007. ISSN: 1879033x, 0959440x. DOI: 10.1016/j.sbi.2007.03.014.

[63] M. Stein, R. R. Gabdoulline, and R. C. Wade. Calculating enzyme kinetic parameters from protein structures. eng. *Biochemical society transactions*, 36(1):51–54, 2008. ISSN: 14708752, 03005127. DOI: 10.1042/BST0360051.

[64] L. M. Tran, M. L. Rizk, and J. C. Liao. Ensemble modeling of metabolic networks. *Biophysical journal*, 95(12):5606–17, Dec. 2008.

[65] J. Wang, W. Wang, P. Kollmann, and D. Case. Antechamber, an accessory software packagefor molecular mechanical calculation. *J. comput. chem.*, 25:1157–1174, 2005.

[66] L. Wang, Y. Deng, J. L. Knight, Y. Wu, B. Kim, W. Sherman, J. C. Shelley, T. Lin, and R. Abel. Modeling Local Structural Rearrangements Using FEP/REST: Application to Relative Binding Affinity Predictions of CDK2 Inhibitors. *Journal of chemical theory and computation*, 9(2):1282–1293, 2013.

[67] M. Waskom, O. Botvinnik, drewokane, P. Hobson, David, Y. Halchenko, S. Lukauskas, J. B. Cole, J. Warmenhoven, J. d. Ruiter, S. Hoyer, J. Vanderplas, S. Villalba, G. Kunter, E. Quintero, M. Martin, A. Miles, K. Meyer, T. Augspurger, T. Yarkoni, P. Bachant, M. Williams, C. Evans, C. Fitzgerald, Brian, D. Wehner, G. Hitz, E. Ziegler, A. Qalieh, and A. Lee. Seaborn: v0.7.1. June 2016.

[68] U. Wittig, R. Kania, M. Golebiewski, M. Rey, L. Shi, L. Jong, E. Algaa, A. Weidemann, H. Sauer-Danzwith, S. Mir, O. Krebs, M. Bittkowski, E. Wetsch, I. Rojas, and W. Müller. SABIO-RK - Database for biochemical reaction kinetics. *Nucleic acids research*, 40(D1):790–796, 2012.

[69] S. Wong, R. E. Amaro, and J. A. McCammon. MM-PBSA Captures Key Role of Intercalating Water Molecules at a ProteinProtein Interface. *Journal of chemical theory and computation*, 5(2):422–429, 2009.

[70] H. Yim, R. Haselbeck, W. Niu, C. Pujol-Baxley, A. Burgard, J. Boldt, J. Khandurina, J. D. Trawick, R. E. Osterhout, R. Stephen, J. Estadilla, S. Teisan, H. B. Schreyer, S. Andrae, T. H. Yang, S. Y. Lee, M. J. Burk, and S. Van Dien. Metabolic engineering of *Escherichia coli* for direct production of 1,4-butanediol. *Nature chemical biology*, 7(7):445–52, July 2011.

[71] M. Yun, C.-G. Park, J.-y. Kim, and H.-w. Park. Structural Analysis of Glyceraldehyde 3-Phosphate Dehydrogenase from *Escherichia coli*: Direct Evidence of Substrate Binding and Cofactor-Induced Conformational Changes. *Biochemistry*, 39:10702–10710, 2000.

[72] X. Zhang, S. E. Wong, and F. C. Lightstone. Towards fully automated high performance computing drug discovery : A massively parallel virtual screening pipeline for docking and MM/GBSA rescoring to improve enrichment. *Journal of chemical information and modeling*, 2013.

Chapter 5

Reaction control is altered on the seconds timescale in pre-steady state glycolysis: a rapid injection NMR study and its computational simulation

Abstract

Dynamic tracking of intracellular reaction cascades is a means of obtaining mechanistic insight into metabolism. We combine time-resolved rapid injection NMR spectroscopy with a computational simulation of the experiment to detect changes of the rate limiting steps occurring during the upstart of glycolysis. Bottlenecks in the glycolytic reaction cascade shift from downstream to upstream reactions within few seconds owing to rapid changes in cofactor pools. Computational systems models of central metabolism appear to have achieved a level of quality that permits correct mechanistic predictions of unanticipated biochemical events.

The model and code used for simulations are available at:

https://github.com/martamatos/Yeast-simulations.

5.1 Introduction

Living systems, and increasingly also designed systems, sense changes in their environment and respond to them. In living systems, these responses occur over a broad timescale. Metabolic responses can occur on the seconds time scale to warrant homeostasis, and fast metabolic reaction cascades such as Embden-Meyerhof-Parnas glycolysis operate on this time scale (Figure 5.1). Failure to respond to dynamically changing environments can lead to loss of fitness, disease or death. Such failed control of Embden-Meyerhof-Parnas glycolysis occurs for instance in cancer and diabetes [9, 24].

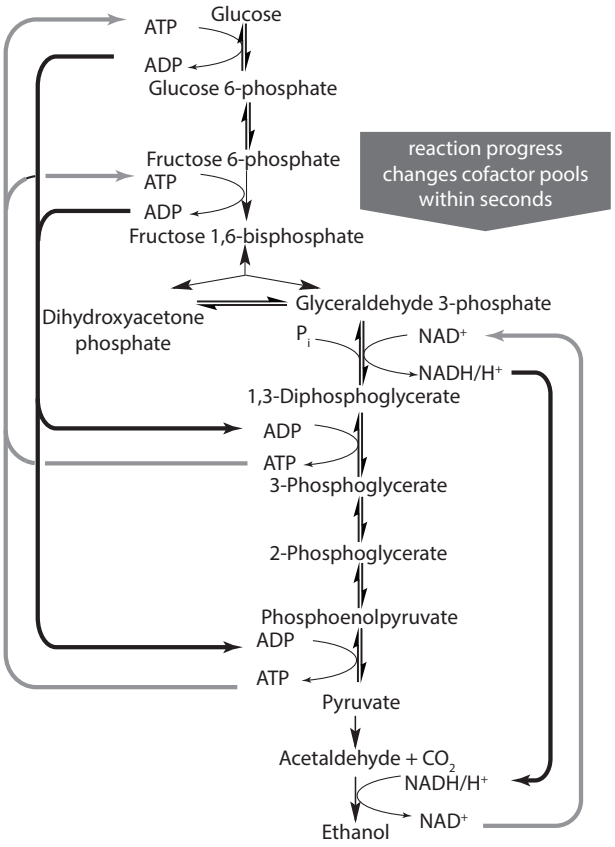


Figure 5.1: Overview of Embden-Meyerhoff-Parnas glycolysis. Reaction progress can affect redox ($NADH/NAD^+$) and energy (ATP/ADP) currencies on the seconds time scale

Unsurprisingly, the mechanistic understanding of robustness and adaptation of intracellular reactions in changeable environments has remained poorly understood due to technical challenges. Resolved signals from the different chemicals and sub-second

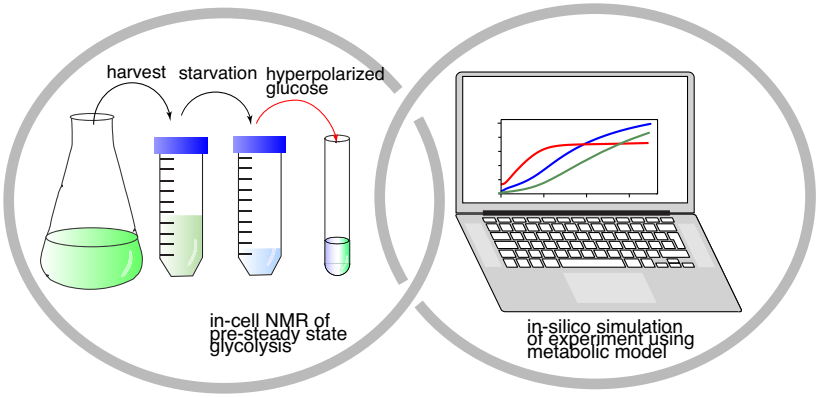


Figure 5.2: Experimental design to track pre-steady state glycolysis using a bolus of hyperpolarized glucose probe in starved cells. Experimental data are rationalized through a computational simulation of the experiment.

time resolution are needed to directly track the reaction cascade with sufficient data points in the living state [3]. Computational models of reaction cascades in glycolysis have been built from experimental Michaelis-Menten characterizations of the isolated catalytic pathway enzymes [21, 7, 20]. In the absence of equivalent experimental data, these models are hard to probe for their predictive value in the living system. NMR spectroscopy can experimentally provide resolved signals from the different chemicals in multistep reaction cascades. A variant of NMR spectroscopy combines chemical labeling with the NMR-visible ^{13}C isotope and a physical labeling approach that redistributes ("hyperpolarizes") [1] nuclear spin states to achieve a short-lived boost of NMR sensitivity. The ^{13}C-labeled hyperpolarized probe molecule improves signals of interest relative to the background 10^5-10^6 fold and makes detection of reactions on the 0.1-100 seconds time scales feasible. Thus, hyperpolarized NMR spectroscopy can provide valuable data to further validate kinetic models for different metabolic pathways.

Hyperpolarized NMR spectroscopy can noninvasively track the dynamic conversion of carbohydrate spins to glycolytic intermediates and end products. The system remains largely underdetermined, however, relative to the number of unknowns, including metabolite concentrations, association constants and the rates of forward and backward reactions in each cascade step. We therefore combine experimental observations of pre-steady state glycolysis in starved cells and the simulation of the experiments using metabolic *in silico* models to identify changes of the rate limiting steps in the upstart of glycolysis (Figure 5.2).

5.2 Results

In cancer cells, bacteria and yeast, dihydroxyacetone phosphate was previously identified as a main intermediate of the glycolytic reaction cascade [13, 14, 8, 5, 17, 12]. Dihydroxyacetone phosphate equilibrates with a minor species of its aldehyde epimer glyceraldehyde 3-phosphate, which is subsequently oxidized and phosphorylated in a reaction catalyzed by the enzyme glyceraldehyde-3-phosphate dehydrogenase (GAPDH).

The accumulation of dihydroxyacetone phosphate is unanticipated from early computational models of yeast glycolysis, but is consistent with experimental determinations of metabolite concentrations at steady state [21]. The kinetic experiments resulted in metabolites where a C-^{2}H rather than a C- ^{1}H bond is broken by GAPDH. Hence, it remained unclear, whether the disagreement between experimental data and early computational models results form experimental isotope effects or from shortcomings of the model [13, 11]. In order to probe if dihydroxyacetone phosphate oxidation is a bottleneck in natural glycolytic reaction cascades, we employed various hyperpolarized substrates that result in protonated positions for dehydrogenase reactions (shown in the Supporting information, Figure 5.6). These experiments with substrates that yield protonated C-1 positions of glyceraldehyde intermediate show that dihydroxyacetone phosphate accumulates as an intermediate in the living system both for subsequent C^{1}H and C^{2}H cleavage. Hence, other factors than CH bond cleavage are limiting for the GAPDH catalyzed conversion of the dihydroxyacetone phosphate intermediate in the living state. Notably, the GAPDH reaction has also been identified as a rate determining step in computational models of cancer glycolysis [19], and in more recent models of yeast glycolysis [7, 20]. Encouraged by the agreement of experimentally observed and computationally predicted bottlenecks, we undertook to study mechanistic responses in pre-steady state glycolysis. This undertaking is further motivated by: (i) the rapid fading of hyperpolarized NMR signal inside living cells [10], implying that changing pathway bottlenecks can be probed in a time-resolved manner (rather than an influx of the probe into cellular steady state pools); and (ii) the fast real time detection of *in vivo* spectra (on the sub-second timescale) allowing pre-steady state *in vivo* measurements [15].

To probe pre-steady state reactions in the upstart of glycolysis, cells were starved for 20 minutes. Upon a glucose pulse applied to starved cell suspensions, 3-phosphoglycerate [23] was initially labeled, alongside minor formation of phosphoenolpyruvate (Figure 5.7). These metabolites are formed downstream of the GAPDH reaction, showing

that the GAPDH reaction is not limiting in starved cells, but only becomes limiting during the upstart of glycolysis. Thus, the rate-determining step in the glycolytic reaction cascade dynamically changes upon an external glucose stimulus within few seconds (Figure 5.3a). The increasingly upstream bottlenecks have conceivable functions in avoiding the exhaustion of cofactors used in upper glycolysis [22].

Experimental data of the detected changes of the rate limiting steps in pre-steady state glycolysis were compared to predictions from a computational simulation of the experiment using a recent kinetic model of *Saccharomyces cerevisiae* glycolysis [20]. The model comprises network topology and the enzyme kinetic parameters obtained under standardized conditions. Simulations of time-dependent metabolite concentrations were conducted using the MASS toolbox in Mathematica® [18]. The yeast metabolic model was used to evolve intracellular metabolite pools in the absence of glucose for 20 minutes to mimic starvation, prior to simulating a glucose bolus (Figure 5.3b). This simulation correctly predicts the initial accumulation of 3-phosphoglycerate, followed by accumulation of dihydroxyacetone phosphate and fructose 1,6-bisphosphate. Thus, experiment and simulation show the same shifts of bottlenecks towards upstream cascade steps on the seconds timescale in pre-steady state glycolysis (Figure 5.3c). However, the *in vivo* kinetics seem to be considerably faster than the *in silico* kinetics, with the measured glycolytic intermediates accumulating faster *in vivo* than *in silico*. This seems to indicate that the enzymes' k_{cat} and/or the active enzyme concentrations are higher *in vivo* than *in silico*. One reason for slower kinetics *in silico* could be that there was a higher than expected inactive amount of enzyme in the assay used to measure the kinetic parameters. This could be due to enzyme instability or simply due to enzymes being in a conformation to which substrates cannot bind [16, 2, 20], possibly due to the purification or assay methods used. Moreover, even though the kinetic parameters were measured in standard conditions defined to resemble physiological conditions, the presence of e.g. certain ions in the media can lead to enzyme activation or inactivation that may or may not happen *in vivo* [6].

The observed slowdown in labeling of 3-phosphoglycerate in pre-steady glycolysis could result from an increasing exhaustion of NAD^+ (Figure 5.1). In order to test the role of the cellular redox state in regulation of glycolysis, an established experimental perturbation was applied, where acetaldehyde is coinjected with glucose [4]. Acetaldehyde is an easily membrane permeable electron acceptor that regenerates NAD^+ for the GAPDH reaction. Accordingly, fluorescence measurements had shown an increase

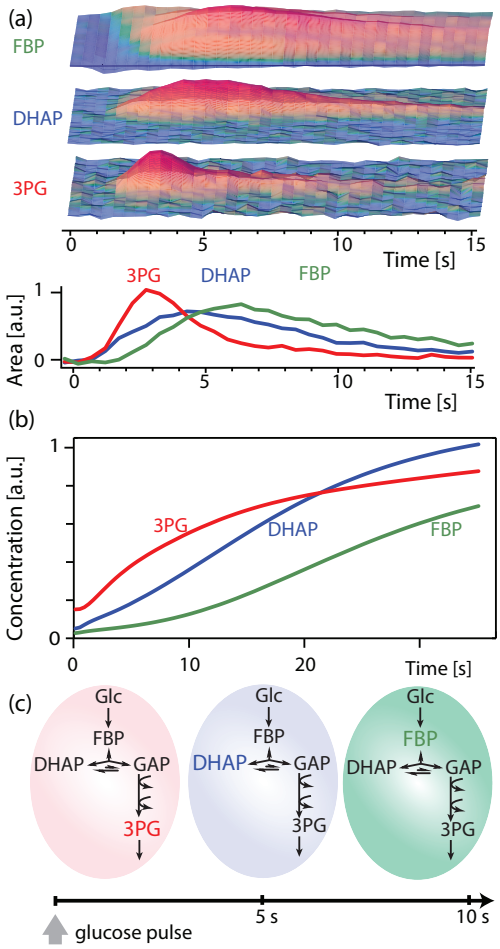


Figure 5.3: (a) Formation of cellular fructose 1,6-bisphosphate (Fbp), dihydroxyacetone phosphate (DHAP) and 3-phosphoglycerate from exogenous glucose. ^{13}C NMR spectroscopic data when using hyperpolarized [U-^{13}C, ^{2}H]glucose are shown (top) in addition to corresponding signal integrals (bottom), showing that labeling of lower glycolytic intermediates precedes labeling of upper intermediates. Signals fade as non-equilibrium hyperpolarization fades. (b) Simulated response of the startup of glycolysis in starved yeast cells. The simulation qualitatively reproduces the trend shown in (a). Changes of the rate limiting steps occur during the startup of glycolysis on the seconds time scale, shifting bottlenecks upstream (c).

of the free cytosolic NAD^+/NADH ratio when coinjecting acetaldehyde with a glucose bolus to yeast cells (2.3 mM of acetaldehyde and glucose), while a drop in the NAD^+/NADH ratio had been measured when injecting glucose without acetaldehyde [4]. Measurements of pre-steady state glycolysis under redox perturbation show that dynamic labeling of 3-phosphoglycerate persists in the presence of electron acceptor. The GAPDH reaction does not become rate limiting within few seconds if NAD^+ is regenerated (Figure 5.4). In the conversion of glucose to ethanol, presence of added acetaldehyde makes acetaldehyde reduction to ethanol limiting due to the exhaustion of NADH (Figure 5.4). In the presence of the reducing carbon source ethanol, effects are opposite to those of added acetaldehyde, resulting in a more severe bottleneck at the GAPDH reaction but less severe bottleneck at the 3-phosphoglycerate isomerization step, consistent with a more reducing cellular milieu in the presence of electron donor. Hence, the bottlenecks of upper and lower glycolysis indirectly probe the cellular redox state (Figure 5.5). Notably, the experimentally observed effects of redox perturbation on accumulation of 3-phosphoglycerate, pyruvate, and dihydroxyacetone phosphate could all be simulated *in silico* (Figures 5.8-5.10).

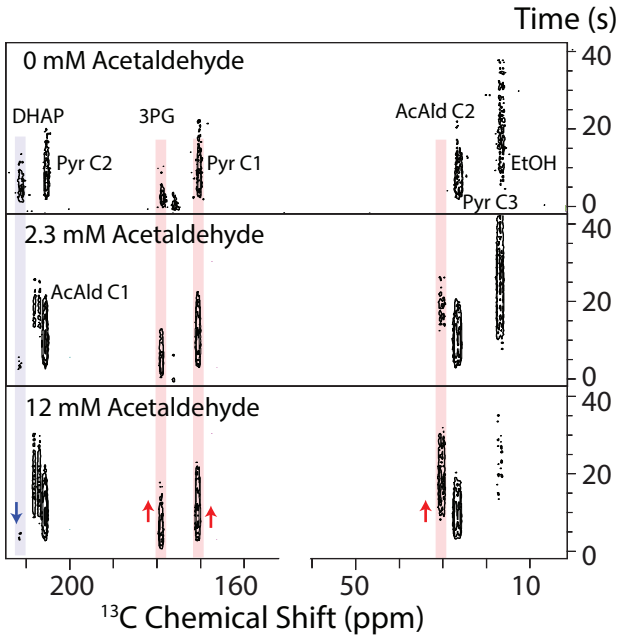


Figure 5.4: Real-time spectroscopy of glucose conversion to intracellular metabolites in response to disturbed redox state. Coninjection of acetaldehyde as an electron acceptor increases dihydroxyacetone phosphate conversion and maintains the downstream reaction of 3-phosphoglycerate as a bottleneck.

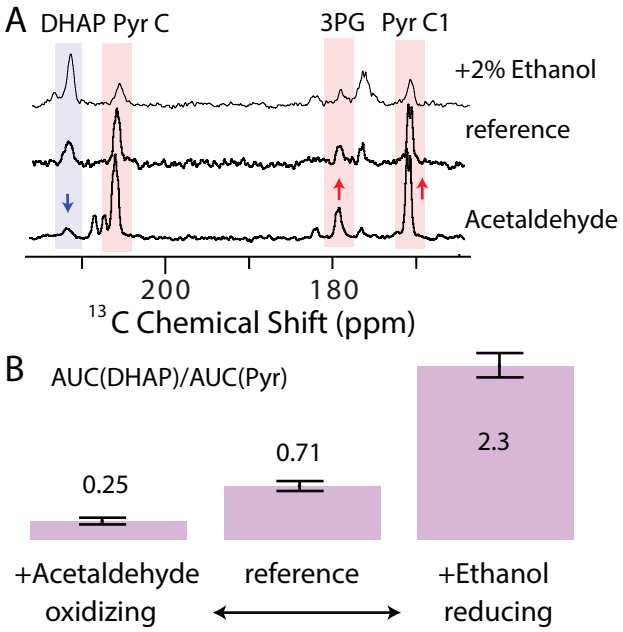


Figure 5.5: Opposing effects of external electron acceptor (addition of acetaldehyde) and electron donor (growth in ethanol) on rate-limiting steps in the glycolytic cascade. (A) Sum spectra of time-resolved rapid-injection experiments show the shifts in cascade bottlenecks, resulting in signal areas that are dependent on redox state (B).

5.3 Final remarks

In conclusion, shifts in rate limiting steps as the cell responds to the availability of glucose can be probed by rapid injection NMR spectroscopy. Metabolites are labeled by nuclear spins deriving from glucose in an order that demonstrates a change of rate-limiting steps in glycolysis within seconds. Self-regulation of the glycolytic cascade sustains flux at low substrate concentrations but avoids an exhaustion of cofactors in upper glycolysis when ample substrate is available. A role of the cellular redox state in the regulation of cascade bottlenecks was deduced by perturbation experiments. Acetaldehyde had been known to elicit dynamic responses in cells, most notably by inducing glycolytic oscillations. Herein, acetaldehyde was shown to shift glycolytic dynamics by maintaining a state that is reminiscent of glycolytic upstart, causing a delayed burst in acetaldehyde formation from glucose substrate. Experimental findings on time- and redox-dependent changes of the rate limiting steps in pre-steady state glycolysis could be qualitatively validated by *in silico* simulations using metabolic models. Thus, systems models of central metabolism appear to have achieved a level of quality that permits correct mechanistic predictions of unanticipated biochemical

events. Inaccuracies in the predictions can be identified experimentally, such as an underestimation of pathway control by GAPDH in earlier models [21] and an overestimation of phosphoenolpyruvate formation in recent models [20]. The combined use of rapid injection NMR spectroscopy in starved cells and *in silico* modeling thus provides a promising method for characterizing cellular metabolism with increasing mechanistic detail.

5.4 Supplementary information

Experimental details

Chemicals

Isotope-enriched metabolic substrates [U-^{13}C, U-^{2}H] glucose, [2-^{13}C]-fructose and [6-^{13}C,6,6'-$^{2}H_2$]-glucose) were purchased from Cortec (Voisins-Le-Bretonneux, France), Cambridge Isotope Laboratories (Tewksbury, MA, USA) and Medical Isotopes Inc. (Pelham, NH, USA). All other chemicals were purchased from Sigma Aldrich (Andover, MA, USA).

Cell growth

Shake flask cultures of *S. cerevisiae* BY4743 (MATa/MATα; his3Δ1/his3Δ1 leu2Δ0/leu2Δ0 lys2Δ0/LYS2 met15Δ0/MET15 ura3Δ0/ura3Δ0) were inoculated in 250 ml Erlenmyer flasks containing 250 mL YPD medium (yeast extract (1% w/v), peptone (2% w/v), glucose (2 % w/v)) from YPD agar plates and incubated over night under shaking at 200 rpm at 30 °C in. These over night precultures were used to inoculate the main yeast cultures of 100 mL YPD (in one-liter Erlenmyer flasks) at a dilution of 1:100, resulting in a OD600 = 0.15–0.20. The shake flask cultures were grown at 30 °C and 200 rpm shaking to OD600 = 0.8. The cultures were harvested by centrifugation in 50 mL Falcon tubes (5000 g, 5 min) and the cell pellet was washed with 20 mL MES buffer (30 mM, pH 5.65). After a second centrifugation the pellet was resuspended in 2.2 mL fresh MES buffer (30 mM, pH 5.65) prior to transfer to a 10 mm NMR tube. Yeast cell suspensions were starved in the absence of added glucose or other carbon source for a total of 20 minutes prior to the in cell NMR experiment. For the in cell NMR experiment, cell suspensions were placed in a 600 MHz Bruker spectrometer 20 min after the last resuspension. In order to evaluate the effect of reducing environment on cellular glycolysis, yeast cells were maintained

for 30 minutes in YPE medium (yeast extract (1% w/v), peptone (2% w/v), ethanol (2 % v/v)) prior to harvest and in cell NMR experiment.

DNP hyperpolarization of hexoses

Solid state DNP polarization of all hexose substrates were performed as previously described using trityl radical OX063 (27 mm; Oxford Instruments, Abingdon, UK) and trimeric gadolinium chelate of 1,3,5-tris-(N-(DO3A-acetamido)-N-methyl-4-amino-2-methylphenyl)-[1,3,5]triazinane-2,4,6-trione (0.9 mm; GE Healthcare) [10, 13]. Samples contained 90 μmol hexose in 19 mg of aqueous polarization medium containing 27 mM trityl radical OX063 and 0.9 mM trimeric Gd chelate. Samples were flash-frozen in liquid helium and polarization transfer was conducted for one hour at 1.2 K by microwave irradiation at 93.89 GHz with 100 mW in a magnetic field of 3.35 T. The solid state polarization buildup was monitored every 5 minutes and solid state polarizations in the self-glassing carbohydrate syrup achieved on the order of 30%. After 1 hour of polarization, the samples were dissolved with heated Milli-Q water (4.5 mL containing EDTA (100 mg/L)) to yield liquid samples with a final substrate concentration of 20 mM hexose. Hyperpolarized substrates (600 μL) were forcefully, manually injected into 2.4 mL of the cell suspension placed inside a 600 MHz Bruker spectrometer. This preparation resulted in a final concentration of 4 mM hyperpolarized carbohydrate substrate for the in-cell NMR experiment.

In cell NMR

All spectra were recorded on a DRX600 spectrometer (Bruker, Karlsruhe, Germany) equipped with a 10 mm BBO probe head that was thermally equilibrated to 30 °C. Glycolytic reaction was followed by a series of ^{13}C NMR spectra that were recorded as pseudo-2D spectra using low flip angle (5º) pulses for excitation. A ^{13}C NMR spectrum of 16384 complex data points was record every 0.5 s. As data acquisition was started prior to substrate injection, experimental dead-time upon substrate feeding was minimized. All spectra were processed in Topspin 3.5 (Bruker) with extensive zero filling and an exponential line broadening of 10 Hz; spectra were integrated in the same software.

NAD$^+$/NADH alteration

Experiments with experimental modulation of the cellular NAD$^+$/NADH ratio were performed as described above with the exception that acetaldehyde at the desired concentration was dissolved in 0.5 mL Milli-Q water and placed in the bottom of the

receiving container for the hyperpolarized hexose in order to achieve a coninjection of acetaldehyde [4] with the received hyperpolarized glucose substrate. Subsequently, in cell NMR experiments were conducted as described above.

In silico modeling

Model simulations

The glycolysis model by Smallbone *et al* [20] was used for the *in silico* modeling of *Saccharomyces cerevisiae* glycolysis, and all simulations were carried out in the MASS toolbox [18] in Mathematica®. Experiments were simulated *in silico* as follows: the cell starvation was simulated by setting the extracellular glucose concentration to 0.1 mM for 20 minutes. The extracellular glucose concentration is defined as a constant in the model utilized here. In order to simulate the glucose pulse, the extracellular concentration of glucose was raised to 4 mM after the 20 minutes starvation period and the evolution of key metabolites and fluxes in yeast glycolysis were simulated for 60 seconds, i.e. the approximate time of the in cell DNP-NMR experiment. In order to simulate the coinjection of glucose and acetaldehyde, the extracellular concentration of glucose was raised to 4 mM and the internal concentration of acetaldehyde was raised to either 2.3 mM or 12 mM after the 20 min starvation period. Simulations correctly predict a rapid drop in cellular NAD^+/NADH ratio upon a glucose pulse, but a rapid increase in NAD^+/NADH upon coinjection of acetaldehyde (see Figures 5.7, 5.8). Likewise, a rapid accumulation of 3-phosphoglycerate, dihydroxyacetone phosphate, and fructose 1,6-bp on the seconds time scale were correctly predicted upon a glucose pulse (Section 5.2, Figure 5.3), as were significantly lower accumulation of fructose-1p and glyceraldehyde 3-phosphate. A comparison between the model and transient NMR data shows that the model appears to overestimate the accumulation of phosphoenolpyruvate. This finding is consistent with reported difficulties in determining and modeling kinetic parameters in the conversion of 3-phosphoglycerate to phosphoenolpyruvate [20] and with previous determinations of much lower phosphoenolpyruvate levels than predicted from the current model [20]. Glycolytic responses upon a modulation of the cellular redox state are consistent between DNP-NMR data of pre-steady glycolysis and simulation, yielding increased accumulation of pyruvate and 3-phosphoglycerate, but lower accumulation of dihydroxyacetone phosphate upon coinjection of acetaldehyde in the upstart of glycolysis (Figure 5.9).

The model

The final mathematical model developed in [20] was used, model 18 in the article. The model is also available at http://identifiers.org/biomodels.db/MODEL1303260018.

The model includes the following (iso)enzymes:

Reaction	Reaction abbreviation	Isoenzymes
acetate branch	acetate branch	
Alcohol dehydrogenase	ADH	Adh1p, Adh5p
Adenylate kinase	AK	-
ATPase	ATPase	-
Enolase	ENO	Eno1p, Eno2p
Fructose-bisphosphate aldolase	FBA	Fba1p
Glyceraldehyde phosphate dehydrogenase	GAPDH	Tdh1p, Tdh2p, Tdh3p
Glycerol 3-phosphate dehydrogenase	GPD	-
Phosphoglyceromutase	GPM	Gpm1p
Glycerol 3-phosphatase	GPP	-
Hexokinase	HXK	Glk1p, Hxk1p, Hxk2p
Glucose transport	HXT	-
Pyruvate decarboxylase	PDC	Pdc1p, Pdc5p, Pdc6p
Phosphofructokinase	PFK	Pfk1p and Pfk2p
Phosphoglucose isomerase	PGI	Pgi1p
3-phosphoglycerate kinase	PGK	Pgk1p
Phosphoglucomutase	PGM	-
Pyruvate kinase	PYK	Cdc19p, Pyk2p
Succinate branch	succinate branch	-
T6P synthase	TPS	-
Triosephosphate isomerase	TPI	Tpi1p
T6P phosphatase	TPP	-
UDP glucose phosphorylase	UGP	-
UDP to UTP pseudoreaction	UDP-to-UTP	-

The corresponding rate laws are:

$$v_{acetate\_branch} = k_{acetate\,branch}[AcAld][NAD]$$

$$v_{ADH1} = \frac{[ADH1]k_{cat,ADH1}\left(\frac{[AcAld][NADH]}{K^{AcAld}_{m,ADH1}K^{NADH}_{i,ADH1}} - \frac{[ETOH][NAD]}{K^{AcAld}_{m,ADH1}K^{NADH}_{i,ADH1}K_{eq,ADH}}\right)}{1 + \frac{[NADH]}{K^{NADH}_{i,ADH1}} + \frac{[AcAld]K^{NADH}_{m,ADH1}}{K^{NADH}_{i,ADH1}K^{AcAld}_{m,ADH1}} + \frac{[ETOH]K^{NAD}_{m,ADH1}}{K^{NAD}_{i,ADH1}K^{ETOH}_{m,ADH1}} + \frac{[NAD]}{K^{NAD}_{i,ADH1}}}$$

$$+ \frac{[AcAld][NADH]}{K^{NADH}_{i,ADH1}K^{AcAld}_{m,ADH1}} + \frac{[NAD][ETOH]K^{NAD}_{m,ADH1}}{K^{NADH}_{i,ADH1}K^{NAD}_{i,ADH1}K^{ETOH}_{m,ADH1}} + \frac{[AcAld][NAD]K^{NADH}_{m,ADH1}}{K^{NADH}_{i,ADH1}K^{NAD}_{i,ADH1}K^{AcAld}_{m,ADH1}}$$

$$+ \frac{[ETOH][NAD]}{K^{NAD}_{i,ADH1}K^{ETOH}_{m,ADH1}} + \frac{[AcAld][NADH][ETOH]}{K^{NADH}_{i,ADH1}K^{ETOH}_{i,ADH1}K^{AcAld}_{m,ADH1}} + \frac{[AcAld][ETOH][NAD]}{K^{AcAld}_{i,ADH1}K^{NAD}_{i,ADH1}K^{ETOH}_{m,ADH1}}$$

$$v_{ADH5} = \frac{[ADH5]k_{cat,ADH5}\left(\frac{[AcAld][NADH]}{K^{AcAld}_{m,ADH5}K^{NADH}_{i,ADH5}} - \frac{[ETOH][NAD]}{K^{AcAld}_{m,ADH5}K^{NADH}_{i,ADH5}K_{eq,ADH}}\right)}{1 + \frac{[NADH]}{K^{NADH}_{i,ADH5}} + \frac{[AcAld]K^{NADH}_{m,ADH5}}{K^{NADH}_{i,ADH5}K^{AcAld}_{m,ADH5}} + \frac{[ETOH]K^{NAD}_{m,ADH5}}{K^{NAD}_{i,ADH5}K^{ETOH}_{m,ADH5}} + \frac{[NAD]}{K^{NAD}_{i,ADH5}}}$$

$$+ \frac{[AcAld][NADH]}{K^{NADH}_{i,ADH5}K^{AcAld}_{m,ADH5}} + \frac{[NAD][ETOH]K^{NAD}_{m,ADH5}}{K^{NADH}_{i,ADH5}K^{NAD}_{i,ADH5}K^{ETOH}_{m,ADH5}} + \frac{[AcAld][NAD]K^{NADH}_{m,ADH5}}{K^{NADH}_{i,ADH5}K^{NAD}_{i,ADH5}K^{AcAld}_{m,ADH5}}$$

$$+ \frac{[ETOH][NAD]}{K^{NAD}_{i,ADH5}K^{ETOH}_{m,ADH5}} + \frac{[AcAld][NADH][ETOH]}{K^{NADH}_{i,ADH5}K^{ETOH}_{i,ADH5}K^{AcAld}_{m,ADH5}} + \frac{[AcAld][ETOH][NAD]}{K^{AcAld}_{i,ADH5}K^{NAD}_{i,ADH5}K^{ETOH}_{m,ADH5}}$$

$$v_{AK} = k_{AK}\left([ADP][ADP] - \frac{[AMP][ATP]}{K_{eq,AK}}\right)$$

$$v_{ATPase} = \frac{V_{max,ATPase}\left(\frac{[ATP]}{K^{ATP}_{m,ATPase}}\right)}{1 + \frac{[ATP]}{K^{ATP}_{m,ATPase}}}$$

$$v_{ENO1} = \frac{[ENO_1]k_{cat,ENO1}\left(\frac{[2PG]}{K^{2PG}_{m,ENO1}} - \frac{[PEP]}{K^{2PG}_{m,ENO1}K_{eq,ENO}}\right)}{1 + \frac{[2PG]}{K^{2PG}_{m,ENO1}} + \frac{[PEP]}{K^{PEP}_{m,ENO1}}}$$

$$v_{ENO2} = \frac{[ENO_2]k_{cat,ENO2}\left(\frac{[2PG]}{K^{2PG}_{m,ENO2}} - \frac{[PEP]}{K^{2PG}_{m,ENO2}K_{eq,ENO}}\right)}{1 + \frac{[2PG]}{K^{2PG}_{m,ENO2}} + \frac{[PEP]}{K^{PEP}_{m,ENO2}}}$$

$$v_{FBA} = \frac{[FBA1]k_{cat,FBA1}\left(\frac{[FBP]}{K^{FBP}_{m,FBA1}} - \frac{[DHAP][G3P]}{K^{FBP}_{m,FBA1}K_{eq,FBA}}\right)}{1 + \frac{[FBP]}{K^{FBP}_{m,FBA1}} + \frac{[DHAP]}{K^{DHAP}_{m,FBA1}} + \frac{[G3P]}{K^{G3P}_{m,FBA1}} + \frac{[FBP][G3P]}{K^{FBP}_{m,FBA1}K^{G3P}_{i,FBA1}} + \frac{[DHAP][G3P]}{K^{DHAP}_{m,FBA1}K^{G3P}_{m,FBA1}}}$$

$$v_{GAPDH1} = \frac{[TDH1]k_{cat,GAPDH1}\left(\frac{[G3P][NAD]}{K^{G3P}_{m,GAPDH1}K^{NAD}_{m,GAPDH1}} - \frac{[13DPG][NADH]}{K^{G3P}_{m,GAPDH1}K^{NAD}_{m,GAPDH1}K_{eq,GAPDH}}\right)}{\left(1 + \frac{[G3P]}{K^{G3P}_{m,GAPDH1}} + \frac{[13DPG]}{K^{13DPG}_{m,GAPDH1}}\right)\left(1 + \frac{[NAD]}{K^{NAD}_{m,GAPDH1}} + \frac{[NADH]}{K^{NADH}_{m,GAPDH1}}\right)}$$

$$v_{GAPDH2} = \frac{[TDH2]k_{cat,GAPDH2}\left(\frac{[G3P][NAD]}{K^{G3P}_{m,GAPDH2}K^{NAD}_{m,GAPDH2}} - \frac{[13DPG][NADH]}{K^{G3P}_{m,GAPDH2}K^{NAD}_{m,GAPDH2}K_{eq,GAPDH}}\right)}{\left(1 + \frac{[G3P]}{K^{G3P}_{m,GAPDH2}} + \frac{[13DPG]}{K^{13DPG}_{m,GAPDH2}}\right)\left(1 + \frac{[NAD]}{K^{NAD}_{m,GAPDH2}} + \frac{[NADH]}{K^{NADH}_{m,GAPDH2}}\right)}$$

$$v_{GAPDH3} = \frac{[TDH3]k_{cat,GAPDH3}\left(\frac{[G3P][NAD]}{K^{G3P}_{m,GAPDH3}K^{NAD}_{m,GAPDH3}} - \frac{[13DPG][NADH]}{K^{G3P}_{m,GAPDH3}K^{NAD}_{m,GAPDH3}K_{eq,GAPDH}}\right)}{\left(1 + \frac{[G3P]}{K^{G3P}_{m,GAPDH3}} + \frac{[13DPG]}{K^{13DPG}_{m,GAPDH3}}\right)\left(1 + \frac{[NAD]}{K^{NAD}_{m,GAPDH3}} + \frac{[NADH]}{K^{NADH}_{m,GAPDH3}}\right)}$$

$$v_{GPD} = \frac{V_{max,GPD}\left(\frac{[DHAP][NADH]}{K^{DHAP}_{m,GPD}K^{NADH}_{m,GPD}} - \frac{[Glyc3P][NAD]}{K^{DHAP}_{m,GPD}K^{NADH}_{m,GPD}K_{eq,GPD}}\right)}{\left(1+\frac{[FBP]}{K^{FBP}_{m,GPD}}+\frac{[ATP]}{K^{ATP}_{m,GPD}}+\frac{[ADP]}{K^{ADP}_{m,GPD}}\right)\left(1+\frac{[DHAP]}{K^{DHAP}_{m,GPD}}+\frac{[Glyc3P]}{K^{Glyc3P}_{m,GPD}}\right)\left(1+\frac{[NADH]}{K^{NADH}_{m,GPD}}+\frac{[NAD]}{K^{NAD}_{m,GPD}}\right)}$$

$$v_{GPM1} = \frac{[GPM1]k_{cat,GPM1}\left(\frac{[3PG]}{K^{3PG}_{m,GPM1}} - \frac{[2PG]}{K^{2PG}_{m,GPM1}K_{eq,GPM1}}\right)}{1+\frac{[3PG]}{K^{3PG}_{m,GPM1}}+\frac{[2PG]}{K^{2PG}_{m,GPM1}}}$$

$$v_{GPP} = \frac{V_{max,GPP}\left(\frac{[Glyc3P]}{K^{Glyc3P}_{m,GPP}}\right)}{1+\frac{[Glyc3P]}{K^{Glyc3P}_{m,GPP}}}$$

$$v_{GLK1} = \frac{[GLKL1]k_{cat,GLK1}\left(\frac{[Glc_{cell}][ATP]}{K^{Glc}_{m,GLK1}K^{ATP}_{m,GLK1}} - \frac{[G6P][ADP]}{K^{Glc}_{m,GLK1}K^{ATP}_{m,GLK1}K_{eq,HXK}}\right)}{\left(1+\frac{[Glc_{cell}]}{K^{Glc}_{m,GLK1}}+\frac{[G6P]}{K^{G6P}_{m,GLK1}}\right)\left(1+\frac{[ATPl]}{K^{ATP}_{m,GLK1}}+\frac{[ADP]}{K^{ADP}_{m,GLK1}}\right)}$$

$$v_{HXK1} = \frac{[HXK1]k_{cat,HXK1}\left(\frac{[Glc_{cell}][ATP]}{K^{Glc_{cell}}_{m,HXK1}K^{ATP}_{m,HXK1}} - \frac{[G6Pl][ADP]}{K^{Glc_{cell}}_{m,HXK1}K^{ATP}_{m,HXK1}K_{eq,HXK}}\right)}{\left(1+\frac{[Glc_{cell}]}{K^{Glc_{cell}}_{m,HXK1}}+\frac{[G6P]}{K^{G6P}_{m,HXK1}}+\frac{[T6P]}{K^{T6P}_{i,HXK1}}\right)\left(1+\frac{[ATP]}{K^{ATP}_{m,HXK1}}+\frac{[ADP]}{K^{ADP}_{m,HXK1}}\right)}$$

$$v_{HXK2} = \frac{[HXK2]k_{cat,HXK2}\left(\frac{[Glc_{cell}][ATP]}{K^{Glc_{cell}}_{m,HXK2}K^{ATP}_{m,HXK2}} - \frac{[G6Pl][ADP]}{K^{Glc_{cell}}_{m,HXK2}K^{ATP}_{m,HXK2}K_{eq,HXK}}\right)}{\left(1+\frac{[Glc_{cell}]}{K^{Glc_{cell}}_{m,HXK2}}+\frac{[G6P]}{K^{G6P}_{m,HXK2}}+\frac{[T6P]}{K^{T6P}_{i,HXK2}}\right)\left(1+\frac{[ATP]}{K^{ATP}_{m,HXK2}}++\frac{[ADP]}{K^{ADP}_{m,HXK2}}\right)}$$

$$v_{HXT} = \frac{V_{max,HXT}\left(\frac{[Glc^{ex}]-[Glc^{cell}]}{K^{Glc}_{m,HXT}}\right)}{1+\frac{[Glc^{ex}]}{K^{Glc}_{m,HXT}}+\frac{[Glc^{cell}]}{K^{Glc}_{m,HXT}}+\frac{\frac{K_{i,HXT}[Glc^{ex}]}{K^{Glc}_{m,HXT}}[Glc^{cell}]}{K^{Glc}_{m,HXT}}}$$

$$v_{PDC1} = \frac{[PDC1]k_{cat,PDC1}\left(\frac{[Pyr]}{K^{Pyr}_{m,PDC1}}\right)}{1+\frac{[Pyr]}{K^{Pyr}_{m,PDC1}}}$$

$$v_{PDC5} = \frac{[PDC5]k_{cat,PDC5}\left(\frac{[Pyr]}{K^{Pyr}_{m,PDC5}}\right)}{1+\frac{[Pyr]}{K^{Pyr}_{m,PDC5}}}$$

$$v_{PDC6} = \frac{[PDC6]k_{cat,PDC6}\left(\frac{[Pyr]}{K^{Pyr}_{m,PDC6}}\right)}{1+\frac{[Pyr]}{K^{Pyr}_{m,PDC6}}}$$

$$min([PFK1],[PFK2])\,k_{cat,PFK}$$

$$gR_{PFK}\frac{[F6P][ATP]}{K^{F6P}_{m,PFK}K^{ATP}_{m,PFK}}\left(1-\frac{\frac{[FBP][ADP]}{[F6P][ATP]}}{K_{eq,PFK}}\right)\left(1+\frac{[F6P]}{K^{F6P}_{m,PFK}}\right.$$

$$v_{PFK} = \frac{\left.+\frac{[ATP]}{K^{ATP}_{m,PFK}}+\frac{\frac{gR_{PFK}[F6P]}{K^{F6P}_{m,PFK}}}{K^{ATP}_{m,PFK}}+\frac{[FBP]}{K^{FBP}_{m,PFK}}+\frac{[ADP]}{K^{ADP}_{m,PFK}}+\frac{\frac{gR_{PFK}[FBP]}{K^{FBP}_{m,PFK}}[ADP]}{K^{ADP}_{m,PFK}}\right.}{\left(1+\frac{[F6P]}{K^{F6P}_{m,PFK}}+\frac{[ATP]}{K^{ATP}_{m,PFK}}+\frac{\frac{gR_{PFK}[F6P]}{K^{F6P}_{m,PFK}}[ATP]}{K^{ATP}_{m,PFK}}+\frac{[FBP]}{K^{FBP}_{m,PFK}}+\frac{[ADP]}{K^{ADP}_{m,PFK}}+\frac{\frac{gR_{PFK}[FBP]}{K^{FBP}_{m,PFK}}}{K^{ADP}_{m,PFK}}\right)^2}$$

$$+L_{0,PFK}\left(\frac{1+\frac{C^{ATP}_{i,PFK}[ATP]}{K^{ATP}_{i,PFK}}}{1+\frac{[ATP]}{K^{ATP}_{i,PFK}}}\right)^2\left(\frac{1+\frac{C^{AMP}_{PFK}[AMP]}{K^{AMP}_{m,PFK}}}{1+\frac{[AMP]}{K^{AMP}_{m,PFK}}}\right)^2\left(\frac{1+\frac{C^{F26BP}_{PFK}[F26BP]}{K^{F26BP}_{m,PFK}}+\frac{C^{FBP}_{PFK}[FBP]}{K^{FBP}_{m,PFK}}}{1+\frac{[F26BP]}{K^{F26BP}_{m,PFK}}+\frac{[FBP]}{K^{FBP}_{m,PFK}}}\right)^2$$

$$\left(1+\frac{C^{ATP}_{PFK}[ATP]}{K^{ATP}_{m,PFK}}\right)^2$$

$$v_{PG11} = \frac{[PGI1]k_{cat,PGI1}\left(\frac{[G6P]}{K^{G6P}_{m,PGI1}} - \frac{[F6P]}{K^{G6P}_{m,PGI1}K_{eq,PGI1}}\right)}{1 + \frac{[G6P]}{K^{G6P}_{m,PGI1}} + \frac{[F6P]}{K^{F6P}_{m,PGI1}}}$$

$$v_{PGK1} = \frac{[PGK1]k_{cat,PGK1}\left(\frac{[ADP]}{K^{ADP}_{m,PGK1}}\right)^{n^{ADP}_{H,PGK}-1}\left(\frac{[13DPG][ADP]}{K^{13DPG}_{m,PGK1}K^{ADP}_{m,PGK1}} - \frac{[3PG][ADP]}{K^{13DPG}_{m,PGK1}K^{ADP}_{m,PGK1}K_{eq,PGK1}}\right)}{\left(1 + \frac{[13DPG]}{K^{13DPG}_{m,PGK1}} + \frac{[3PG]}{K^{3PG}_{m,PGK1}}\right)\left(1 + \left(\frac{[ADP]}{K^{ADP}_{m,PGK1}}\right)^{n^{ADP}_{H,PGK}} + \frac{[ATP]}{K^{ATP}_{m,PGK1}}\right)}$$

$$v_{PGM} = \frac{V_{max,PGM}\left(\frac{[G6P]}{K^{G6P}_{m,PGM}} - \frac{[G1P]}{K^{G6P}_{m,PGM}K_{eq,PGM}}\right)}{1 + \frac{[G6P]}{K^{G6P}_{m,PGM}} + \frac{[G1P]}{K^{G1P}_{m,PGM}}}$$

$$v_{CDC19} = \frac{[CDC19]k_{cat,CDC19}\left(\frac{[PEP][ADP]}{K^{PEP}_{m,CDC19}K^{ADP}_{m,CDC19}} - \frac{[Pyr][ATP]}{K^{PEP}_{m,CDC19}K^{ADP}_{m,CDC19}K_{eq,PYK}}\right)}{\left(1 + \frac{[PEP]}{K^{PEP}_{m,CDC19}} + \frac{[Pyr]}{K^{Pyr}_{m,CDC19}} + L_{0,CDC19}\frac{\frac{[ATP]}{K^{ATP}_{i,CDC19}}+1}{\frac{[FBP]}{K^{FBP}_{m,CDC19}}+1}\right)\left(1 + \frac{ADP}{K^{ADP}_{m,CDC19}} + \frac{[ATP]}{K^{ATP}_{m,CDC19}}\right)}$$

$$v_{PYK2} = \frac{[PYK2]k_{cat,PYK2}\left(\frac{[PEP][ADP]}{K^{PEP}_{m,PYK2}K^{ADP}_{m,PYK2}} - \frac{[Pyr][ATP]}{K^{PEP}_{m,PYK2}K^{ADP}_{m,PYK2}K_{eq,PYK}}\right)}{\left(1 + \frac{[PEP]}{K^{PEP}_{m,PYK2}} + \frac{[Pyr]}{K^{Pyr}_{m,PYK2}} + L_{0,PYK2}\frac{\frac{[ATP]}{K^{ATP}_{i,PYK2}}+1}{\frac{[FBP]}{K^{FBP}_{m,PYK2}}+1}\right)\left(1 + \frac{ADP}{K^{ADP}_{m,PYK2}} + \frac{[ATP]}{K^{ATP}_{m,PYK2}}\right)}$$

$$v_{succinate\_branch} = k_{succinate\_branch}[Pyr][NAD]$$

$$v_{TPS} = \frac{V_{max,TPS}\left(\frac{[G6P][UDP_{glc}]}{K^{G6P}_{m,TPS}K^{UDP_{glc}}_{m,TPS}}\right)}{\left(1 + \frac{[G6P]}{K^{G6P}_{m,TPS}}\right)\left(1 + \frac{[UDP_{glc}]}{K^{UDP_{glc}}_{m,TPS}}\right)}$$

$$v_{TPI1} = \frac{[TPI1]k_{cat,TPI}\left(\frac{[DHAP]}{K_{m,TPI1}^{DHAP}} - \frac{[G3P]}{K_{m,TPI1}^{DHAP}K_{eq,TPI}}\right)}{1 + \frac{[DHAP]}{K_{m,TPI1}^{DHAP}} + \frac{[G3P]}{K_{m,TPI1}^{G3P}}\left(1 + \left(\frac{[G3P]}{K_{i,TPI}^{G3P}}\right)^4\right)}$$

$$v_{TPP} = \frac{V_{max,TPP}\left(\frac{[T6P]}{K_{m,TPP}^{T6P}}\right)}{1 + \frac{[T6P]}{K_{m,TPP}^{T6P}}}$$

$$v_{UGP} = \frac{V_{max,UGP}\left(\frac{[UTP][G1P]}{K_{m,UGP}^{UTP}K_{m,UGP}^{G1P}}\right)}{\frac{K_{i,UGP}^{UTP}}{K_{m,UGP}^{UTP}} + \frac{[UTP]}{K_{m,UGP}^{UTP}} + \frac{[G1P]}{K_{m,UGP}^{G1P}} + \frac{[UTP][G1P]}{K_{m,UGP}^{UTP}+K_{m,UGP}^{G1P}} + \frac{\frac{K_{i,UGP}^{UTP}}{K_{m,UGP}^{UTP}}[UDP_{glc}]}{K_{i,UGP}^{UDP_{glc}}} + \frac{[G1P][UDP_{glc}]}{K_{m,UGP}^{G1P}K_{i,UGP}^{UDP_{glc}}}}$$

$$v_{UDP-to-UTP} = k_{UDP-to-UTP}[UDP][ATP]$$

The system of Ordinary Differential Equations is the following:

$$\frac{d([ADP])}{dt} = v_{UDP-to-UTP} - v_{PYK2} - v_{CDC19} - v_{PGK1} + v_{PFK} + v_{HXK2} + v_{HXK1} + v_{GLK1} + v_{ATPase} - 2 \cdot v_{AK}$$

$$\frac{d([ATP])}{dt} = -v_{UDP-to-UTP} - +v_{PYK2} + v_{CDC19} + v_{PGK1} - v_{PFK} - v_{HXK2} - v_{HXK1} - v_{GLK1} - v_{ATPase} + 2 \cdot v_{AK}$$

$$\frac{d([AcAld])}{dt} = v_{PDC6} + v_{PDC5} + v_{PDC1} - v_{acetate\,branch} - v_{ADH5} - v_{ADH1}$$

$$\frac{d([13DPG])}{dt} = v_{TDH1} - v_{PGK1} + v_{GAPDH2} + v_{GAPDH3}$$

$$\frac{d([DHAP])}{dt} = v_{FBA} - v_{TPI} - v_{GPD}$$

$$\frac{d([FBP])}{dt} = v_{PFK} - v_{FBA}$$

$$\frac{d([F6P])}{dt} = v_{PGI} - v_{PFK}$$

$$\frac{d([G1P])}{dt} = v_{PGM} - v_{UDG}$$

$$\frac{d([Glyc3P])}{dt} = -v_{GPP} + v_{GPD}$$

$$\frac{d([G6P])}{dt} = -v_{PGM} - v_{PGI} + v_{HXK2} + v_{HXK1} + v_{GLK1} - v_{TPS}$$

$$\frac{d([G3P])}{dt} = -v_{GAPDH1} - v_{GAPDH2} - v_{GAPDH3} + v_{FBA} + v_{TPI}$$

$$\frac{d([Glc_{cell}])}{dt} = v_{HXT} - v_{HXK1} - v_{HXK2} - v_{GLK1}$$

$$\frac{d([NAD])}{dt} = -v_{GAPDH1} - v_{GAPDH2} - v_{GAPDH3} - 3 \cdot v_{succinate\,branch} + v_{GPD} - v_{acetate\,branch} + v_{ADH5} + v_{ADH1}$$

$$\frac{d([2PG])}{dt} = v_{GPM1} - v_{ENO1} - v_{ENO2}$$

$$\frac{d([3PG])}{dt} = v_{PGK1} - v_{GPM1}$$

$$\frac{d([PEP])}{dt} = -v_{PYK2} - v_{CDC19} + v_{ENO1} + v_{ENO2}$$

$$\frac{d([Pyr])}{dt} = v_{PYK2} + v_{CDC19} - v_{PDC1} - v_{PDC5} - v_{PDC6} - v_{succinate\,branch}$$

$$\frac{d([T6P])}{dt} = v_{TPS} - v_{TPP}$$

$$\frac{d([UDP])}{dt} = -v_{UDP-to-UTP} + v_{TPSe}$$
$$\frac{d([UTP])}{dt} = v_{UDP-to-UTP} - v_{UDG}$$

To this system of ODEs the following mass conservation equations are added:

$$[AMP] = sum_{AXP} - [ATP] - [ADP]$$
$$[NADH] = sum_{NAD} - [NAD]$$
$$[UDP_{glc}] = sum_{UXP} - [UTP] - [UDP]$$
$$energy\_charge = \frac{[ATP] + \frac{[ADP]}{2}}{sum_{AXP}}$$

where:

$$sum_{AXP} = [AMP] + [ADP] + [ATP] = 6.02$$
$$sum_{NAD} = [NAD] + [NADH] = 1.59$$
$$sum_{UXP} = [UDP_{glc}] + [UDP] + [UTP] = 1.4$$

The enzyme concentrations used are:

Isoenzyme	Concentration (mM/L)
ADH1	0.164
ADH5	0.0042
CDC19	2.05
ENO1	0.686
ENO2	1.97
FBA1	1.34
GLK1	0.045
GPD1	0.0068
GPD2	0.00079
GPM1	0.73
HOR	0.0055
HXK1	0.017
HXK2	0.061
PDC1	1.07
PDC5	0.012
PDC6	0.0065
PFK1	0.047
PFK2	0.039
PGI1	0.14
PGK1	0.26
PGM1	0.0033
PGM2	0.0013
PYK2	0.0061
RHR2	0.051
TDH1	0.35
TDH2	0
TDH3	4.2
TPI1	0.29
TPS1	0.0034
TPS2	0.0027

The initial metabolite concentrations are:

Metabolite	Concentration (mM)	Is the concentration constant?
13DPG	0.00074	No
2PG	0.0677	No
3PG	0.47	No
AcAld	0.178	No
Acetate	223	Yes
ADP	1.29	No
ATP	4.29	No
DHAP	1.16	No
ETOH	221.9	Yes
FBP	4.58	No
F26bP	0.003	Yes
F6P	0.24	No
G1P	0.54	No
G6P	0.77	No
Glc_{cell}	6.28	No
G3P	0.32	No
Glycerol	0.15	Yes
Glyc3P	0.27	No
NAD	1.5	No
PEP	0.61	No
Pyr	2.1	No
Succinate	0	Yes
Trehalose	0.015	Yes
T6P	0.02	No
UDP	0.28	No
UGP1	0.0062	Yes
UTP	0.65	No

The parameter values are:

Reaction	Isoenzyme	Parameter	Value
Acetate branch	-	k	0.0055 s^{-1}
ADH	-	K_{eq}	14492.8
ADH	Adh1p	k_{cat}	176 s^{-1}
ADH	Adh1p	K_m^{ETOH}	17 mM
ADH	Adh1p	K_i^{ETOH}	90 mM
ADH	Adh1p	K_m^{AcAld}	0.4622 mM
ADH	Adh1p	K_i^{AcAld}	1.1 mM
ADH	Adh1p	K_m^{NAD}	0.17 mM
ADH	Adh1p	K_i^{NAD}	0.92 mM
ADH	Adh1p	K_m^{NADH}	0.11 mM
ADH	Adh1p	K_i^{NADH}	0.031 mM
ADH	Adh5p	k_{cat}	0 s^{-1}
ADH	Adh5p	K_m^{ETOH}	17 mM
ADH	Adh5p	K_i^{ETOH}	90 mM
ADH	Adh5p	K_m^{AcAld}	1.11 mM
ADH	Adh5p	K_i^{AcAld}	1.1 mM
ADH	Adh5p	K_m^{NAD}	0.17 mM
ADH	Adh5p	K_i^{NAD}	0.92 mM
ADH	Adh5p	K_m^{NADH}	0.11 mM
ADH	Adh5p	K_i^{NADH}	0.031 mM
AK	-	K_{eq}	0.45
AK	-	k	0.75 s^{-1}
ATPase	-	V_{max}	6.16 mM/s
ATPase	-	K_m^{ATP}	3 mM
ENO	-	K_{eq}	6.7
ENO	Eno1p	k_{cat}	7.6 s^{-1}
ENO	Eno1p	K_m^{2PG}	0.043 mM
ENO	Eno1p	K_m^{PEP}	0.5 mM
ENO	Eno2p	k_{cat}	19.9 s^{-1}
ENO	Eno2p	K_m^{2PG}	0.104 mM
ENO	Eno2p	K_m^{2PG}	0.5 mM

FBA	-	K_{eq}	0.069
FBA	Fba1p	k_{cat}	4.14 s^{-1}
FBA	Fba1p	K_m^{FBP}	0.451 mM
FBA	Fba1p	K_m^{G3P}	2.4 mM
FBA	Fba1p	K_i^{G3P}	10 mM
FBA	Fba1p	K_m^{DHAP}	2 mM
GAPDH	-	K_{eq}	0.0053
GAPDH	Tdh1p	k_{cat}	19.1 s^{-1}
GAPDH	Tdh1p	K_m^{G3P}	0.495 mM
GAPDH	Tdh1p	K_m^{13DPG}	0.0098 mM
GAPDH	Tdh1p	K_m^{NAD}	0.09 mM
GAPDH	Tdh1p	K_m^{NADH}	0.06 mM
GAPDH	Tdh2p	k_{cat}	8.63 s^{-1}
GAPDH	Tdh2p	K_m^{G3P}	0.77 mM
GAPDH	Tdh2p	K_m^{13DPG}	0.0098 mM
GAPDH	Tdh2p	K_m^{NAD}	0.09 mM
GAPDH	Tdh2p	K_m^{NADH}	0.06 mM
GAPDH	Tdh3p	k_{cat}	18.2 s^{-1}
GAPDH	Tdh3p	K_m^{G3P}	0.423 mM
GAPDH	Tdh3p	K_m^{13DPG}	0.909 mM
GAPDH	Tdh2p	K_m^{NAD}	0.09 mM
GAPDH	Tdh2p	K_m^{NADH}	0.06 mM
GPD	-	K_{eq}	10000
GPD	-	V_{max}	0.78 mM/s
GPD	-	K_m^{ADP}	2 mM
GPD	-	K_m^{ATP}	0.73 mM
GPD	-	K_m^{DHAP}	0.54 mM
GPD	-	K_m^{FBP}	4.8 mM
GPD	-	K_m^{glyc3P}	1.2 mM
GPD	-	K_m^{NAD}	0.93 mM
GPD	-	K_m^{NADH}	0.023 mM
GPP	–	V_{max}	0.883 mM/s
GPP	-	K_m^{G3P}	3.5 mM
GPM	-	K_{eq}	0.19
GPM	Gpm1p	k_{cat}	400 s^{-1}
GPM	Gpm1p	K_m^{2PG}	1.41 mM

GPM	Gpm1p	K_m^{3PG}	1.2 mM
HXK	-	K_{eq}	2000
HXK	Glk1	k_{cat}	0.0721 s^{-1}
HXK	Glk1	K_m^{Glc}	0.0106 mM
HXK	Glk1	K_m^{ATP}	0.865 mM
HXK	Glk1	K_m^{ADP}	0.23 mM
HXK	Glk1	K_m^{G6P}	30 mM
HXK	Hxk1p	k_{cat}	10.2 s^{-1}
HXK	Hxk1p	K_m^{Glc}	0.15 mM
HXK	Hxk1p	K_m^{ATP}	0.293 mM
HXK	Hxk1p	K_m^{ADP}	0.23 mM
HXK	Hxk1p	K_m^{G6P}	30 mM
HXK	Hxk1p	K_i^{T6P}	0.2 mM
HXK	Hxk2p	k_{cat}	63.1 s^{-1}
HXK	Hxk2p	K_m^{Glc}	0.2 mM
HXK	Hxk2p	K_m^{ATP}	0.195 mM
HXK	Hxk2p	K_m^{ADP}	0.23 mM
HXK	Hxk2p	K_m^{G6P}	30 mM
HXK	Hxk2p	K_i^{T6P}	0.04 mM
HXT	-	K_m^{glc}	0.9 mM
HXT	-	K_i	0.91 mM
HXT	-	V_{max}	3.35 mM/s
PDC	Pdc1p	k_{cat}	12.1 s^{-1}
PDC	Pdc1p	K_m^{PYR}	8.5 mM
PDC	Pdc5p	k_{cat}	10.3 s^{-1}
PDC	Pdc5p	K_m^{PYR}	7.08 mM
PDC	Pdc6p	k_{cat}	9.21 s^{-1}
PDC	Pdc6p	K_m^{PYR}	2.92 mM
PFK	-	K_{eq}	800
PFK	Pfk1:Pfk2	k_{cat}	210 s^{-1}
PFK	Pfk1:Pfk2	C_{amp}	0.0845 mM
PFK	Pfk1:Pfk2	C_{atp}	3 mM
PFK	Pfk1:Pfk2	C_{FBP}	0.397 mM
PFK	Pfk1:Pfk2	C_{F26BP}	0.0174 mM
PFK	Pfk1:Pfk2	C_i^{ATP}	100 mM
PFK	Pfk1:Pfk2	K_m^{ADP}	1 mM

PFK	Pfk1:Pfk2	K_m^{AMP}	0.0995 mM
PFK	Pfk1:Pfk2	K_m^{ATP}	0.71 mM
PFK	Pfk1:Pfk2	K_m^{FBP}	0.111 mM
PFK	Pfk1:Pfk2	K_m^{F26BP}	0.000682 mM
PFK	Pfk1:Pfk2	K_m^{F6P}	0.1 mM
PFK	Pfk1:Pfk2	K_i^{ATP}	0.65 mM
PFK	Pfk1:Pfk2	L_0	0.66
PFK	Pfk1:Pfk2	gR	5.12
PGI	-	K_{eq}	0.29
PGI	Pgi1p	k_{cat}	487 s^{-1}
PGI	Pgi1p	K_m^{G6P}	1.03 mM
PGI	Pgi1p	K_m^{F6P}	0.307 mM
PGK	Pgk1p	k_{cat}	58.6 s^{-1}
PGK	Pgk1p	K_{eq}	3200
PGK	Pgk1p	K_m^{3PG}	4.58 mM
PGK	Pgk1p	K_m^{ATP}	1.99 mM
PGK	Pgk1p	K_m^{13DPG}	0.003 mM
PGK	Pgk1p	K_m^{ADP}	0.2 mM
PGK	Pgk1p	n_H^{ADP}	2
PGM	-	K_{eq}	0.1667
PGM	-	V_{max}	0.128 mM/s
PGM	-	K_m^{G1P}	0.023 mM
PGM	-	K_m^{G6P}	0.05 mM
PYK	-	K_{eq}	6500
PYK	Cdc19p	k_{cat}	20.1 s^{-1}
PYK	Cdc19p	K_m^{PEP}	0.281 mM
PYK	Cdc19p	K_m^{ADP}	0.243 mM
PYK	Cdc19p	K_m^{PYR}	21 mM
PYK	Cdc19p	K_m^{ATP}	1.5 mM
PYK	Cdc19p	K_m^{FBP}	0.2 mM
PYK	Cdc19p	K_i^{ATP}	9.3 mM
PYK	Cdc19p	L_0	100
PYK	Pyk2p	k_{cat}	0 s^{-1}
PYK	Pyk2p	K_m^{PEP}	0.19 mM
PYK	Pyk2p	K_m^{ADP}	0.3 mM
PYK	Pyk2p	K_m^{PYR}	21 mM

PYK	Pyk2p	K_m^{ATP}	1.5 mM
PYK	Pyk2p	K_m^{FBP}	0.2 mM
PYK	Pyk2p	K_i^{ATP}	9.3 mM
PYK	Pyk2p	L_0	100
succinate branch	-	k	0 s^{-1}
T6P phosphatase	-	V_{max}	2.34 mM/s
T6P phosphatase	-	K_m^{T6P}	0.5 mM
T6P synthase	-	V_{max}	0.49 mM/s
T6P synthase	-	K_m^{G6P}	3.8 mM
T6P synthase	-	K_m^{UDG}	0.886 mM
TPI	-	k_{cat}	0.045 mM
TPI	Tpi1p	k_{cat}	564 s^{-1}
TPI	Tpi1p	K_m^{DHAP}	6.45 mM
TPI	Tpi1p	K_m^{G3P}	5.25 mM
TPI	Tpi1p	K_i^{G3P}	35.1 mM
UGP	-	V_{max}	13.26 mM/s
UGP	-	K_m^{G1P}	0.32 mM
UGP	-	K_i^{UDG}	0.0035 mM
UGP	-	K_i^{UTP}	0.11 mM
UGP	-	K_m^{UTP}	0.11 mM
UDP-to-UTP	-	k	0.075 s^{-1}

5.5 Supplemental figures

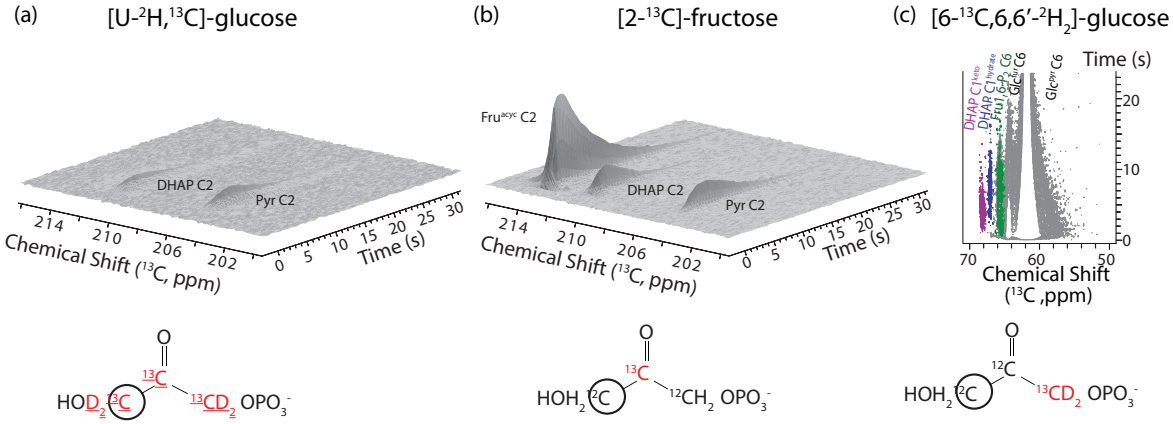


Figure 5.6: In-cell experiments showing that dihydroxyacetone phosphate accumulates as reaction intermediate using various molecular probes. Dihydroxyacetone phosphate formed from [2-^{13}C]-fructose and [6-^{13}C,6,6'-^{2}H$_2$]-glucose is not protonated at the position that is dehydrated in the glyceraldehyde-3-phosphate dehydrogenase reaction (encircled position). GAPDH is thus a limiting step under physiological conditions and other factors than C-H cleavage appear to limit reaction dihydroxyacetone phosphate conversion.

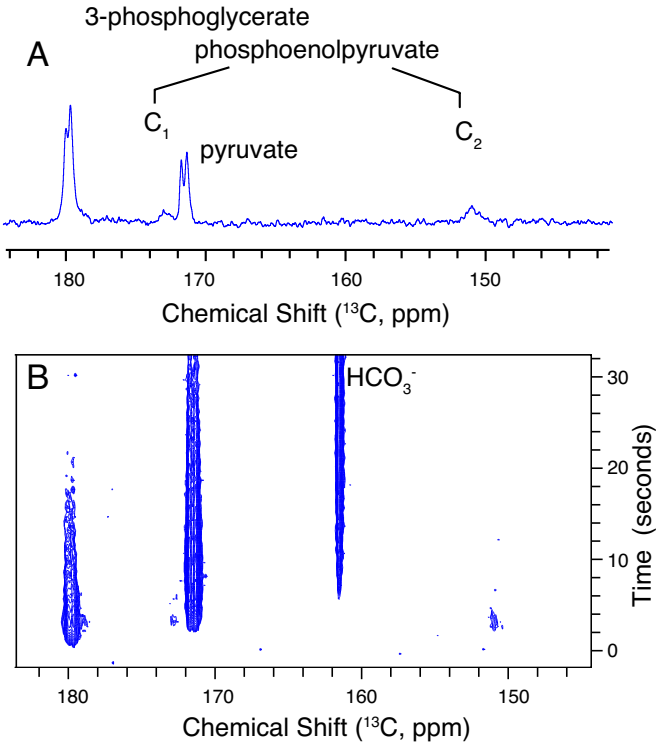


Figure 5.7: Intracellular metabolites formed in *Saccharomyces cerevisiae* upon a pulse of hyperpolarized glucose. Signals formed within the first 5 seconds (A) show that 2-phosphoglycerate predominates under the up-start of glycolysis, with smaller amounts of 2-phosphoenolpyruvate forming concurrently (B), real-time series of ^{13}C NMR spectra). The concurrent emergence of 2-phosphoenolpyruvate and 3-phosphoglycerate is consistent with an equilibrium between mono-phosphorylated C3-acids, while their initial accumulation is consistent with an abundance of NAD^+ and lack of ADP during pre-steady state glycolysis. Minor formation of 2-phosphoenolpyruvate during pre-steady state glycolysis is consistent with steady state metabolite levels, while its formation is overestimated by some recent metabolic models [20].

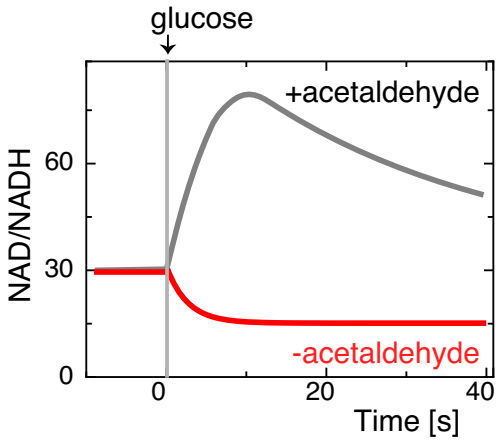


Figure 5.8: Schematic depiction of *Saccharomyces cerevisiae* cytosolic free NAD^+/NADH ratios in response to a 2.3 mM glucose pulse in the presence (grey) and in the absence (red) of 2.3 mM acetaldehyde coinjection, as determined by Canelas *et al* [4] and assuming a pH near 6.45 in the vacuole of non-growing yeast.

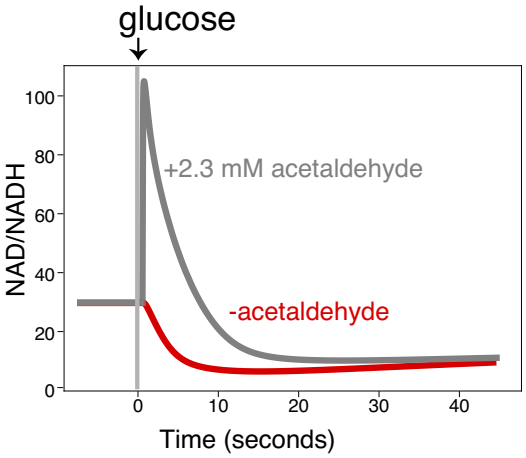


Figure 5.9: Predicted evolution of intracellular NAD^+/NADH ratio in response to a 4 mM glucose pulse in the presence (grey) and in the absence (red) of 2.3 mM acetaldehyde coinjection.

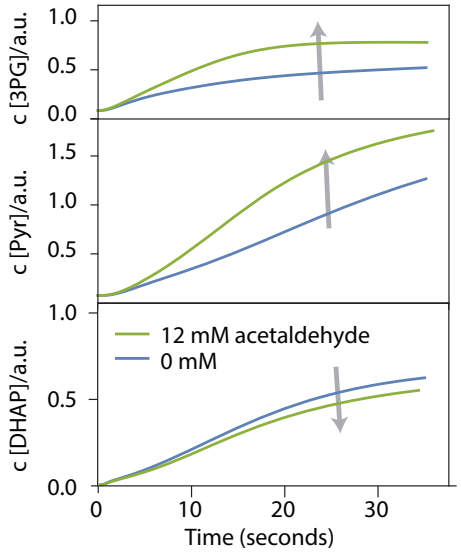


Figure 5.10: Predicted trends for the effects of acetaldehyde coinjection during startup of glycolysis, suggesting increases of 3-phosphoglycerate and pyruvate formation, but a decrease of dihydroxyacetone phosphate, consistent with experimental data shown in Figures 5.4 and 5.5.

References

[1] J. H. Ardenkjaer-Larsen, B. Fridlund, A. Gram, G. Hansson, L. Hansson, M. H. Lerche, R. Servin, M. Thaning, and K. Golman. Increase in signal-to-noise ratio of > 10,000 times in liquid-state NMR. *Proceedings of the national academy of sciences*, 100(18):10158–10163, 2003.

[2] A. Bar-Even, R. Milo, E. Noor, and D. S. Tawfik. The moderately efficient enzyme: futile encounters and enzyme floppiness. eng. *Biochemistry*, 54(32):4969–4977, 2015. ISSN: 15204995, 00062960. DOI: 10.1021/acs.biochem.5b00621.

[3] J. M. Buescher, M. R. Antoniewicz, L. G. Boros, S. C. Burgess, H. Brunengraber, C. B. Clish, R. J. DeBerardinis, O. Feron, C. Frezza, B. Ghesquiere, E. Gottlieb, K. Hiller, R. G. Jones, J. J. Kamphorst, R. G. Kibbey, A. C. Kimmelman, J. W. Locasale, S. Y. Lunt, O. D. Maddocks, C. Malloy, C. M. Metallo, E. J. Meuillet, J. Munger, K. Nöh, J. D. Rabinowitz, M. Ralser, U. Sauer, G. Stephanopoulos, J. St-Pierre, D. A. Tennant, C. Wittmann, M. G. Vander Heiden, A. Vazquez, K. Vousden, J. D. Young, N. Zamboni, and S. M. Fendt. A roadmap for interpreting ^{13}C metabolite labeling patterns from cells. *Current opinion in biotechnology*, 34:189–201, 2015.

[4] A. B. Canelas, W. M. Van Gulik, and J. J. Heijnen. Determination of the cytosolic free NAD/NADH ratio in *Saccharomyces cerevisiae* under steady-state and highly dynamic conditions. *Biotechnology and bioengineering*, 100(4):734–743, 2008.

[5] C. E. Christensen, M. Karlsson, J. R. Winther, P. R. Jensen, and M. H. Lerche. Non-invasive in-cell determination of free cytosolic [NAD^+]/ [NADH] ratios using hyperpolarized glucose show large variations in metabolic phenotypes. *Journal of biological chemistry*, 289(4):2344–2352, 2014.

[6] K. v. Eunen and B. M. Bakker. The importance and challenges of in vivo-like enzyme kinetics. eng. *Perspectives in science*, 1(1-6):126–130, 2014. ISSN: 22130209. DOI: 10.1016/j.pisc.2014.02.011.

[7] K. v. Eunen, J. A. L. Kiewiet, H. V. Westerhoff, and B. M. Bakker. Testing Biochemistry Revisited: How *In Vivo* Metabolism Can Be Understood from *In Vitro* Enzyme Kinetics. *Plos computational biology*, Jan. 2012.

[8] T. Harris, H. Degani, and L. Frydman. Hyperpolarized ^{13}C NMR studies of glucose metabolism in living breast cancer cell cultures. *Nmr in biomedicine*, 26(12):1831–1843, 2013.

[9] J. H. v. Heerden, M. T. Wortel, F. J. Bruggeman, J. J. Heijnen, Y. J. M. Bollen, R. Planque, J. Hulshof, T. G. O'Toole, S. A. Wahl, and B. Teusink. Lost in Transition: Start-Up of Glycolysis Yields Subpopulations of Nongrowing Cells. *Science*, 343(6174):1245114–1245114, 2014.

[10] M. Karlsson, P. R. Jensen, J. H. Ardenkjaer-Larsen, and M. H. Lerche. Difference between Extra- and Intracellular T1 Values of Carboxylic Acids Affects the Quantitative Analysis of Cellular Kinetics by Hyperpolarized NMR. *Angewandte chemie - international edition*, 55(43):13567–13570, 2016.

[11] K. R. Keshari and D. M. Wilson. *Chemistry and biochemistry of ^{13}C hyperpolarized magnetic resonance using dynamic nuclear polarization*. Vol. 43(5), 2014, pp. 1627–1659. ISBN: 1415353166.

[12] M. H. Lerche, P. R. Jensen, M. Karlsson, and S. Meier. NMR insights into the inner workings of living cells. *Analytical chemistry*, 87(1):119–132, 2015.

[13] S. Meier, P. R. Jensen, and J. O. Duus. Real-time detection of central carbon metabolism in living *Escherichia coli* and its response to perturbations. *Febs letters*, 585(19):3133–3138, 2011.

[14] S. Meier, M. Karlsson, P. R. Jensen, M. H. Lerche, and J. Ø. Duus. Metabolic pathway visualization in living yeast by DNP-NMR. *Molecular biosystems*, 7(10):2834, 2011.

[15] S. Meier, M. Karlsson, and P. R. Jensen. Detecting Elusive Intermediates in Carbohydrate Conversion: A Dynamic Ensemble of Acyclic Glucose-Catalyst Complexes. *Acs sustainable chemistry and engineering*, 5(6):5571–5577, 2017.

[16] D. Purich. *Contemporary enzyme kinetics and mechanism: reliable lab solutions*. Of *Selected Methods in Enzymology*. Elsevier Science, 2009. ISBN: 9780123847447.

[17] T. B. Rodrigues, E. M. Serrao, B. W. C. Kennedy, D.-E. Hu, M. I. Kettunen, and K. M. Brindle. Magnetic resonance imaging of tumor glycolysis using hyperpolarized ^{13}C-labeled glucose. *Nature medicine*, 20(1):93–97, 2013.

[18] A. Sastry and N. Sonnenschein. MASS-Toolbox: Introducing Annotations and Pathway Attributes. Jan. 2016. URL: https://doi.org/10.5281/zenodo.44339.

[19] A. A. Shestov, X. Liu, Z. Ser, A. A. Cluntun, Y. P. Hung, L. Huang, D. Kim, A. Le, G. Yellen, J. G. Albeck, and J. W. Locasale. Quantitative determinants of aerobic glycolysis identify flux through the enzyme GAPDH as a limiting step. *Elife*, 3(July2014):1–18, 2014.

[20] K. Smallbone, H. L. Messiha, K. M. Carroll, C. L. Winder, N. Malys, W. B. Dunn, E. Murabito, N. Swainston, J. O. Dada, F. Khan, P. Pir, E. Simeonidis, I. Spasić, J. Wishart, D. Weichart, N. W. Hayes, D. Jameson, D. S. Broomhead, S. G. Oliver, S. J. Gaskell, J. E. G. McCarthy, N. W. Paton, H. V. Westerhoff, D. B. Kell, and P. Mendes. A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *Febs letters*, 587(17):2832–41, Sept. 2013.

[21] B. Teusink, J. Passarge, C. Reijenga, E. Esgalhado, C. v. d. Weijden, M. Schepper, M. Walsh, B. Bakker, K. v. Dam, H. Westerhoff, and J. Snoep. Can yeast glycolysis be understood in terms of *in vitro* kinetics of the constituent enzymes? testing biochemistry. eng. *European journal of biochemistry*, 267(17):5313–5329, 2000.

[22] B. Teusink, M. C. Walsh, K. Van Dam, and H. V. Westerhoff. The danger of metabolic pathways with turbo design. *Trends in biochemical sciences*, 23(5):162–169, 1998.

[23] K. N. Timm, J. Hartl, M. A. Keller, D.-E. Hu, M. I. Kettunen, T. B. Rodrigues, M. Ralser, and K. M. Brindle. Hyperpolarized [U-^{2}H, U-^{13}C] Glucose Reports on Glycolytic and Pentose Phosphate Pathway Activity in EL4 Tumors and Glycolytic Activity in Yeast Cells. *Magnetic resonance in medicine*, 74(6):1543–1547, 2015.

[24] O. Warburg. ÜBER DEN STOFFWECHSEL DER CARCINOMZELLE. *Naturwissenschaften*, 12(50):1131–1137, 1924.

Chapter 6

Discussion and future work

In this thesis we focused on kinetic models for metabolism. We first aimed at developing an approach to build mechanistic kinetic models while making as little assumptions as possible. Towards this goal, in chapter 2, we start by developing a highly flexible software package to parameterize enzymatic reactions individually (MASSef). With MASSef we decompose the enzymatic reaction into its elementary steps and use mass action rate laws to model each step. This results in different sets of elementary rate constants $\boldsymbol{k}$ that can reproduce the kinetic data (e.g. K_m, k_{cat}, K_i) used to fit the model equally well. Hence, we end up not with a single model for each enzyme, but a model ensemble.

Building enzyme-level kinetic models using MASSef has a key limitation. It requires kinetic data that is not always available and is usually hard to find for less studied organisms. On the other hand, the software allows one to fit different kinetics, such as Michaelis-Menten or Hill kinetics, simply by specifying the correct mechanism. Also, it can fit different kinetic parameter values measured under different conditions, and, in principle, it can highlight incompatibilities between different kinetic parameters, which lead to a higher sum of squared log-deviations.

Future work on MASSef

While working on MASSef, we found that the solution previously implemented to model allosteric enzymes, and based on the mechanism described in [9], was not able to reproduce the sigmoidal kinetics measured experimentally in [2]. This solution is based on the Monod-Wyman-Changeux (MWC) model, which assumes that the enzyme can be in two different conformations: one where substrate affinity is lower, termed the tense (T) state, and another where the substrate affinity is higher, termed the relaxed (R) state. A key assumption of the MWC formalism is that, at any given

time point, all enzyme protomers are in the same form, i.e. either all protomers are in the R state or in the T state. The mechanism proposed in [9] defines an extra catalytic track per allosteric site in the enzyme, where a catalytic track is the minimal set of elementary reactions that converts substrates into products in a given reaction. For instance, if the enzyme has four allosteric sites, there will be one catalytic track with one effector bound, another with two effectors bound, and so on. If the effector is an inhibitor, upon binding the enzyme transitions into the T state and it is assumed that the reaction cannot proceed. See Figure 6.1 for a detailed description. This mechanism does not reproduce sigmoidal kinetics though.

$$\boldsymbol{r_{PFK}}\text{: } \boldsymbol{ATP + F6P \rightleftharpoons ADP + F16bP}$$

A

$$PFK_0 + ADP \rightleftharpoons PFK_1$$
$$PFK_1 + ADP \rightleftharpoons PFK_2$$
$$PFK_2 + ADP \rightleftharpoons PFK_3$$
$$PFK_3 + ADP \rightleftharpoons PFK_4$$

B

$$PFK_0 + PEP \rightleftharpoons PFK_1$$
$$PFK_1 + PEP \rightleftharpoons PFK_2$$
$$PFK_2 + PEP \rightleftharpoons PFK_3$$
$$PFK_3 + PEP \rightleftharpoons PFK_4$$

C

$$PFK_0 \rightleftharpoons PFK_0$$

D

$$PFK_i + ATP \rightleftharpoons PFK_i\&ATP$$
$$PFK_i\&ATP + F6P \rightleftharpoons PFK_i\&ATP\&F6P$$
$$PFK_i\&ATP\&F6P \rightleftharpoons PFK_i\&ADP\&F16bP$$
$$PFK_i\&ADP\&F16bP \rightleftharpoons PFK_i\&ADP + F16bP$$
$$PFK_i\&ADP \rightleftharpoons PFK_i + ADP$$

Figure 6.1: Original mechanism for allosteric enzymes, here exemplified for PFK. ADP is an activator, PEP is an inhibitor. Green stands for the R state of the enzyme, while red stands for the T state of the enzyme. The subscript i in PFK_istands for the number of effector molecules bound to the enzyme. (A) Specifies all reactions for activator binding/release; (B) specifies all the reactions for inhibitor binding/release; (C) specifies the spontaneous transition reaction between the R and T state; and (D) specifies all catalytic reactions, where substrates bind to the enzyme R state and are converted into products which are then released. When the enzyme is in the T state no catalytic reactions occur.

A key assumption in this mechanism is that all enzyme protomers have a similar behavior, which is not necessarily true. A more general, and possibly more accu-

rate, mechanism is presented in [10], where all enzyme protomers are modeled and all possible combinations of substrate/effector binding and product/effector release are considered. However, this mechanism results in more elementary reactions and respective rate laws than MASSef, and in particular Mathematica®, can handle to generate the flux equations. Hence, we need to come up with a reasonable approximated mechanism to model sigmoidal kinetics. The reason we do not simply use the generalized generalized MWC expression used in [7, 10] is that it does not distinguish free and total metabolite concentrations.

In the future we will try an approach similar to the one in [13], where the flux equation is divided into a catalytic and a regulatory part:

$$v = f_{catalytic} \cdot f_{regulatory}$$

where

$$f_{catalytic} = n \cdot f(\mathbf{x}_M, \mathbf{k}_R)$$

and $f(\mathbf{x}_M, \mathbf{k}_R)$ is generated from the enzyme mechanism following the usual workflow for a single protomer, n is the number of enzyme protomers, $\mathbf{x}_M$ represents the reaction's substrates and products, and $\mathbf{k}_R$ represents the kinetic parameters when the enzyme is in the R form. The regulatory part will be based on the generalized MWC model [11], which has the general form

$$f_{regulatory} = \frac{1 + (f_T(\mathbf{x}_M, \mathbf{k}_T)/f_R(\mathbf{x}_M, \mathbf{k}_R)) \cdot Q(L, \mathbf{x}_M, \mathbf{x}_E, \mathbf{k}_R, \mathbf{k}_T, \mathbf{k}_E)}{1 + Q(L, \mathbf{x}_M, \mathbf{x}_E, \mathbf{k}_R, \mathbf{k}_T, \mathbf{k}_E)}$$

where $\mathbf{x}_E$ represents the enzyme effectors, $\mathbf{k}_T$ represents the kinetic parameters when the enzyme is in the T form, $\boldsymbol{k}_E$ represents the kinetic parameters for the effectors, L is the allosteric constant that represents the ratio of the enzyme concentration in the T form and the R form, and Q is the function that describes the transition between the R and T conformations and vice-versa. The idea would be to define f_T, f_R and Q using the workflow in MASSef, where f_T and f_R can already be generated in a straightforward way, the challenge lying in the Q function.

Other features that will be soon implemented in MASSef are: 1) the simulation of data to be fitted for enzyme activation based on [4](section 6.7), when activation

constants are available; 2) the automatic addition of the relevant elementary reaction for substrate inhibition and data simulation based on [4](section 6.9.2); 3) the correction for chemical activities using a Debye-Huckel model.

In Chapter 3, we used the enzyme-level kinetic models built through MASSef to build a prototype system-level model ensemble consisting of eight reactions in *E. coli*'s glycolysis. We took two different approaches to build the system-level model, which we named eMASS and eMASS2. In both approaches we drop the assumption $x_{free} \approx x_{tot}$. Following eMASS we first build fully functional enzyme-level models by integrating fluxomics and metabolomics data with each enzyme-level model individually, and only then define the system-level model as the union of all enzyme-level models. With eMASS2, we take the steady-state flux equations derived for each enzymatic reaction, define a constraint on metabolite conservation $x_{tot,i} = \sum_j x_{bound,i,j} + x_{free,i}$ for each metabolite i, and globally sample total enzyme and free metabolite concentrations that fit the fluxomics and metabolomics data. With both approaches we verified that $x_{free} \approx x_{tot}$ does not always hold, as in some models the concentration of e.g. free PEP was only 60% of the respective total concentration. Comparing both approaches, eMASS2 is a more correct way to build a kinetic model as the total enzyme concentrations are calculated based on free metabolite concentrations which are fitted globally. Besides, eMASS2 model simulations can be initialized with steady-state conditions and indeed be at steady-state at $t = 0$ s, unlike eMASS. Furthermore models built through eMASS2 lead to more reasonable predictions of total enzyme concentrations.

The fact that eMASS2 tends to overestimate total enzyme concentrations might be because the model encompasses only a tiny fraction of *E. coli*'s metabolism. This most likely results in overestimated predictions of free metabolite concentrations, which, in turn, can lead to overestimation of total enzyme concentrations, as these depend on free metabolite concentrations. However, the expansion of the prototype model ensemble generated through eMASS2 is dependent on MASSef's ability to model allosteric enzymes. Yet, there will be some technical challenges associated with expanding eMASS/eMASS2. Using eMASS2 will mean longer running times to globally fit the free metabolite and total enzyme concentrations. Currently, with 6561 models it takes $\approx$ 3 hours running in parallel on 122 CPU cores (Intel Xeon Processor E5-2680 v2 – 10 cores, 2.80GHz, 25MB L3 Cache, 128GB RAM).

However, once eMASS/eMASS2 can be used to model, e.g. the central carbon metabolism of *E. coli*, these approaches should lead to more accurate predictions.

This would be particularly useful to design new strains for biotechnological applications.

Finally, it would be interesting to assess the impact of integrating x_{free} vs. x_{tot} in models based on Michaelis-Menten or convenience kinetics rate laws. What would be the effect in model predictions? Given the results presented in [6] and our own preliminary results, the impact might not be critical for most metabolites. However, for metabolites with lower concentrations, such as NADH it might be enough to improve predictions. In [6] the authors hypothesize the difference in NADH concentration prediction vs. measured to be precisely because the model predicts free metabolite while total metabolite is the measured quantity.

In Chapter 4, we tackled the consistent lack of data that has been plaguing kinetic models throughout the years. Using molecular dynamics methods, we estimated the difference in binding energies, $\Delta\Delta G$, between the reaction's product(s) and substrate(s), and integrated this $\Delta\Delta G$ with the respective enzyme-level kinetic model. Here, we found metalloenzymes to be challenging, Still, we showed that, in terms of the number of elementary rate constant combinations allowed in each model ensemble, it can be as constraining as any other parameter. However, the impact of including this $\Delta\Delta G$ at the system-level remains to be assessed. This will be part of future work, as both GAPD and ENO are already part of the model prototypes generated through eMASS/eMASS2. In particular, it would be interesting to assess the impact of integrating $\Delta\Delta G$ in different enzyme models in a given pathway.

The general workflow developed in this project can be applied to any kinetic model that decomposes each reaction into elementary steps. Hence, providing a promising option to parameterize kinetic models and mitigate the lack of kinetic data. Furthermore, the same approach can be applied to include the effect of enzyme mutations in kinetic models.

In Chapter 5, we used a published kinetic model for the glycolysis pathway in *Saccharomyces cerevisiae* [14] to study the cellular response measured by hyperpolarized NMR spectroscopy to: a) a glucose pulse, b) a glucose + acetaldehyde pulse, and c) a glucose + ethanol pulse. The model predictions were in agreement with the experimental results, and confirmed that the observed bottlenecks were due to a change in the NAD^+/NADH ratio, a result which is also consistent with the literature, thus showing a practical application of a kinetic model. We also tried earlier to reproduce these results using earlier models of *S. cerevisiae* glycolysis [15, 12], however, these were not as successful. Thus emphasizing the importance of measuring the kinetic

parameters each enzyme in standard conditions and modeling substrate-level enzyme regulation.

Future work on analyzing hyperpolarized NMR spectroscopy data

In this project, an initial attempt was made to fit the kinetic model for glycolysis in *Saccharomyces cerevisiae* to the measured time-course data. Yet, after several failed attempts and a more careful analysis, we realized a new approach was needed. Since the cells grow on unlabeled glucose until the starvation period of 20 min and are fed labeled glucose only after this starvation period, to successfully model the cellular response to a labeled glucose pulse the model would need to take into account both labeled and unlabeled metabolites. This is because only labeled metabolites can be measured, but unlabeled metabolites also contribute to the system's observable dynamics. Furthermore, after some time the labeled metabolites decay into unlabeled metabolites, a behavior that also needs to be modeled.

Two approaches can, in principle, be taken towards modeling both labeled and unlabeled metabolites in the cell. One approach would be to simply use eMASS2 to build a model considering two versions of each metabolite, a labeled version and an unlabeled version. We would assume that, in a two substrates and two products reaction, both substrates may be either labeled or unlabeled or one substrate may be labeled and the other unlabeled, and likewise for products. We would assume cofactors to be unlabeled. Another, approximate, approach would be to use the original model's rate laws. Here we would again consider both a labeled and an unlabeled version of each metabolite, except for cofactors, and assume two types of reactions: r_l, involving only labeled metabolites, and r_u, involving only unlabeled metabolites. These reactions would need to be coupled such that

$$V_{max} \geq v_l + v_u$$

where v_l represents the flux through the reaction that involves only labeled metabolites, and v_u represents the flux through the reaction that involves only unlabeled metabolites. Since

$$V_{max} \propto [E_{total}]$$

we assume

$$\begin{aligned} v_l &\propto \alpha[E_{total}] \\ v_u &\propto \beta[E_{total}] \end{aligned}$$

where $\alpha = [E_l]/[E_{total}]$ and $[E_l]$ is the enzyme concentration bound to labeled metabolites, $\beta = [E_l]/[E_{total}]$ and $[E_u]$ is the enzyme concentration bound to unlabeled metabolites. Finally,

$$\begin{aligned}[E_{total}] &= [E_l] + [E_u] + [E_{free}] \\ &= \alpha[E_{total}] + \beta[E_l] + \gamma[E_u]\end{aligned}$$

where $\gamma = [E_{free}]/[E_{total}]$. An approximate form for α and β could, in principle, be found by modeling the same system with MASSef but considering only two catalytic tracks, one involving only labeled metabolites and the other only unlabeled metabolites.

While the first approach is, in principle, easier to implement and should lead to more accurate results, the second might turn out to be more practical, even though find a form for α and β might prove challenging.

Overall the integration of computational models and experimental data is a powerful way to study biological processes. In an iterative cycle, experiments provide data to build computational models, which are then used to make predictions to be tested experimentally. Mismatches between predictions and experimental results often lead to new knowledge and improved models. Kinetic models are key to understand the metabolism inherent dynamics. These models have been applied not only in strain design for biotechnological applications [1, 8], but also to build personalized models and identify individuals susceptible to drug side effects [3], as well as to find potential drug targets [5]. Briefly, two main approaches have been taken to build kinetic models: bottom-up, relying on enzyme kinetic parameters, and top-down, relying mostly on omics data. Both approaches have their strengths and weaknesses and, to some extent, kinetics models end up being a combination of both. Yet, a general lack of data has been a major bottleneck to build kinetic models. Here, we developed a new workflow to build mechanistic kinetic models that bridges both bottom-up and top-down approaches: we use historically measured enzyme-level data to build a model ensemble which is then constrained by integrating recently measured omics data, potentially leading to more accurate models. We further addressed data scarcity by developing a workflow to use molecular dynamics to parameterize kinetic models. The same workflow can be applied to model enzyme mutations and be used to study disease.

References

[1] J. Almquist, M. Cvijovic, V. Hatzimanikatis, J. Nielsen, and M. Jirstrand. Kinetic models in industrial biotechnology - Improving cell factory performance. *Metabolic engineering*, 24C:38–60, 2014.

[2] D. Blangy, H. Buc, and J. Monod. Kinetics of the allosteric interactions of phosphofructokinase from *Escherichia coli*. *Journal of molecular biology*, 31(1):13–35, 1968.

[3] A. Bordbar, D. McCloskey, D. C. Zielinski, N. Sonnenschein, N. Jamshidi, and B. O. Palsson. Personalized Whole-Cell Kinetic Models of Metabolism for Discovery in Genomics and Pharmacodynamics. *Cell systems*, 1(4):283–292, 2015.

[4] A. Cornish-Bowden. *Fundamentals of Enzyme Kinetics*. Wiley-Blackwell, 2012.

[5] J. R. Haanstra, A. Gerding, A. M. Dolga, F. J. H. Sorgdrager, M. Buist-Homan, F. d. Toit, K. N. Faber, H.-G. Holzhütter, B. Szöör, K. R. Matthews, J. L. Snoep, H. V. Westerhoff, and B. M. Bakker. Targeting pathogen metabolism without collateral damage to the host. *Scientific reports*, 7(August 2016):40406, 2017.

[6] D. Kesten, U. Kummer, S. Sahle, and K. Hübner. A new model for the aerobic metabolism of yeast allows the detailed analysis of the metabolic regulation during glucose pulse. *Biophysical chemistry*, 206:40–57, 2015.

[7] P. Millard, K. Smallbone, and P. Mendes. Metabolic regulation is sufficient for global and robust coordination of glucose uptake, catabolism, energy production and growth in *Escherichia coli*. *Plos computational biology*, 13(2):e1005396, 2017.

[8] L. Miskovic and V. Hatzimanikatis. Production of biofuels and biochemicals: In need of an ORACLE. *Trends in biotechnology*, 28(8):391–397, 2010.

[9] B. Ø. Palsson. *Systems Biology: Simulation of Dynamic Network States*. Cambridge University Press, 2011. ISBN: 9781139495424.

[10] K. Peskov, I. Goryanin, and O. Demin. Kinetic model of phosphofructokinase-1 from *Escherichia coli*. *Journal of bioinformatics and computational biology*, 6(4):843–867, 2008.

[11] S. V. Popova and E. E. Sel'kov. Generalization of the model by monod, wyman and changeux for the case of a reversible monosubstrate reaction. *Febs letters*, 53(3):269–273, 1975.

[12] L. Pritchard and D. B. Kell. Schemes of flux control in a model of *Saccharomyces cerevisiae* glycolysis. *European journal of biochemistry*, 269(16):3894–3904, Aug. 2002.

[13] P. Saa and L. K. Nielsen. A general framework for thermodynamically consistent parameterization and efficient sampling of enzymatic reactions. *Plos computational biology*, 11(4):e1004195, 2015.

[14] K. Smallbone, H. L. Messiha, K. M. Carroll, C. L. Winder, N. Malys, W. B. Dunn, E. Murabito, N. Swainston, J. O. Dada, F. Khan, P. Pir, E. Simeonidis, I. Spasić, J. Wishart, D. Weichart, N. W. Hayes, D. Jameson, D. S. Broomhead, S. G. Oliver, S. J. Gaskell, J. E. G. McCarthy, N. W. Paton, H. V. Westerhoff, D. B. Kell, and P. Mendes. A model of yeast glycolysis based on a consistent kinetic characterisation of all its enzymes. *Febs letters*, 587(17):2832–41, Sept. 2013.

[15] B. Teusink, J. Passarge, C. Reijenga, E. Esgalhado, C. v. d. Weijden, M. Schepper, M. Walsh, B. Bakker, K. v. Dam, H. Westerhoff, and J. Snoep. Can yeast glycolysis be understood in terms of *in vitro* kinetics of the constituent enzymes? testing biochemistry. eng. *European journal of biochemistry*, 267(17):5313–5329, 2000.